ON-LINE MAINTENANCE TESTS COMPLETED
FRIDAY, 3/ 2/79, 3:51 PM.

LABEL 000000000LINE 001790A1CC EX ONLINE/MAINT;COMMON=21;END←

ONLINE /MAINT

Moore Business Forms, Inc. 7744 11-2

```
 LABEL  000000000LINE   001790&1CC EX 0/R}COMMON=1}FILE S=PATCHES/PASCAL}FILE LINE=LINE PRINT}END+        0       /R

                        FILE:  PATCHES/PASCAL     FRIDAY  03/02/79  03:52 PM

$# PATCH   1 FOR PASCAL.XVT.0 CONTAINS 10 CARDS.  CORRECT SPELLING & TABULATION
$: PATCH TO CORRECT SPELLING IN SOME ERROR MESSAGES, CORRECT TABULATION OF CODE
$: OR COMMENTS, AND TO CORRECT THE CALL ON THE PROCEDURE TO GIVE A NEW PAGE.
$: *** NOTE THAT ERROR(71) IS NOW NO LONGER USED - SEE PATCH  513.
$:     IS WAS USED ONCE, BUT INCORRECTLY. ERROR(63) IS CALLED IN ITS PLACE.
$: *** NOTE THAT THE ALGOL CODE FILE "PASCRUN"/"DISK" HAS BEEN RENAMED
$:     "PASCAL"/"PRELUDE".  IT IS NO LONGER REFERENCED DIRECTLY IN THIS COMPILER
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
    BEGIN     } % NULL                      %***   4) REWRITE                50203000
      GEN("PUT",3,5)}                       %***   5) PAGE                   50204000
      GEN("PPAGE",5,3)}   %                                                  50208000
        BLOCK}                             %*** COMPILE PROCEDURE BODY ***   80646000
    COMPSTAT}                              %*** COMPILE STATEMENT PART ***   80691000
      (" 41  ALFA CONSTANTS MAY NOT BE LONGER THAN 7 CHARACTERS."),          91045000
      (" 55  PROCEDURE NESTING DEPTH + NO OF RECORDS IS TOO GREAT."),        91060000
      (" 87  END-OF-INPUT ENCOUNTERED UNEXPECTEDLY."),                       91094000
      (" 97  TOO MANY FILES IN USE."),                                       91104000
END OF B5700 PASCAL COMPILER.. .....................................99001000
$# PATCH 2 FOR PASCAL CONTAINS 171 CARDS.
$:    PATCH TO MERGE DAG LANGMYHRS PPP10 TO PPP11 COSY PATCHES
$:    WITH NILS OTTES MODIFIED PPP10 SOURCE.
$:    DAVID A COOPER , HERIOT-WATT UNIVERSITY,  JANUARY 1978.
$:
FILE CARD "SOURCE" (1,10,30)}                  % SOURCE CODE FILE          10035000
FILE LINES 1 (1,17)}                           % PRINT FILE                10036000
FILE PASCALGOL DISK SERIAL [20,600] (1,10,30,SAVE 0)} % CODE FILE          10037000
ARRAY PARAMTAB, FORWPARAM1, FORWPARAM2[0:MAXPARAMS]}                       10109000
FILE XREFFILE DISK SERIAL [20:3000] (1,3,30)}                              10137000
ALPHA ARRAY XBUFF[0:2]}                                                    10138500
BOOLEAN XINB}                                                              10138550
INTEGER ARRAY SYMKIND[0:62]}                %USED IN ERROR RECOVERY.       10149000
      ERROR100MESS (//"100  ILLEGAL SAVE CONSTANT IN ""S""" OPTION10188500
. THE VALUE 07 IS SUBSTITUTED";"     SO THIS ERROR DOES NOT INCREMENT TH10188600
E COMPILATION ERRORS COUNT."//),%                                          10188700
        PACKEDSY=61#,  ASSERTSY=62#}                                       10211000
%                                                                          20181500
%                                                                          20181550
    IF ERRNUM=100                                                          20181600
    THEN NUMERRS:=NUMERRS-1;%    * ERROR NUMBER 100 ALONE SHOULD NOT       20181650
%                                * PREVENT THE XALGOL COMPILATION BEING    20181700
%                                * ZIPPED AS THE VALUE 7 IS SUBSTITUTED    20181750
%                                * FOR A BAD SAVE CONSTANT IN AN "S"       20181800
%                                * OPTION.                                 20181850
%                                                                          20181900
%                                                                          20181950
     7(INITIAL),MIDDLE,INITIAL}                                            20308000
        "40OREAD", "6READLN", "50RESET", "6UNPACK", "50WRITE",             20373000
        "600JZXL" DO                                                       20373500
    IF DECL THEN AX := -AX}                                                20520000
    ABS(A[2]) LEQ ABS(B[2]))}                                              20539000
%                                                                          20541100
%                                                                          20541150
```

```
%
BOOLEAN PROCEDURE XREFINPUT(A);                                          20541200
ARRAY A[0];                                                              20541250
BEGIN                                                                    20541300
    LABEL EOF;                                                           20541350
    INTEGER I;                                                           20541400
%                                                                        20541450
    READ(XREFFILE,3,XBUFF[*1)[EOF];                                      20541500
    FOR I:=0,1,2 DO                                                      20541550
     A[I] := XBUFF[I];                                                   20541600
    IF FALSE THEN EOF: BEGIN                                             20541650
                           CLOSE(XREFFILE,RELEASE);                      20541700
                           XINB := TRUE;                                 20541750
                       END;                                              20541800
    XREFINPUT := XINB;                                                   20541850
%                                                                        20541900
END OF XREFINPUT;                                                        20541950
        A2 := -A2;                                                       20541960
BOOLEAN LPARFOUND,SAVEXREFOPT;                                           20570000
    SAVEXREFOPT := XREFOPTION;    XREFOPTION := FALSE;                   20842000
            IF SAVEXREFOPT THEN NEWXREF(CURNAME1,CURNAME2,THISLEVEL,     20847500
                                        FALSE);                          20861500
    XREFOPTION := SAVEXREFOPT;                                           20861550
%         ASSERT              62        ASSERTSY              INITIAL    20868500
                IF CURNAME1="ASSERT" THEN ASSERTSY ELSE                  30075500
            END%                                                         30165500
%                                                                        30280000
%                                                                        30280025
%                                                                        30280050
% THE FOLLOWING    LINES DECODE ANY OCCURRENCE OF THE "S" OPTION AND     30280075
% SETS THE GLOBAL INTEGER VARIABLE "SAVEFACTOR" WHICH CONTROLS THE        30280100
% TYPE OF COMPILATION INITIATED BY THE ZIP. THERE ARE THREE LEGAL FORMS   30280125
% OF THE "S" OPTION AS FOLLOWS:-                                         30280150
%                                                                        30280175
%         "S-"  WILL GIVE NO ZIP IE. PASCAL SYNTAX CHECK ONLY            30280200
%         "S+"  WILL GIVE A ZIP FOR COMPILE AND GO                       30280225
%         "S??" WILL GIVE A ZIP FOR COMPILE TO LIBRARY                   30280250
%               WHERE ?? IS THE TWO DIGIT DECIMAL SAVE                   30280275
%               CONSTANT GIVEN THE OBJECT CODE FILE                      30280300
%               NB. IF THE SAVE CONSTANT IS TO BE                        30280325
%                   LESS THAN 10 THE FIRST DIGIT                         30280350
%                   MUST BE INCLUDED IE. A "0".                          30280375
%                                                                        30280400
%                                                                        30280425
            ELSE                                                         30280450
            IF CX="S" THEN                                               30280475
            BEGIN                                                        30280500
                IF C="-" THEN SAVEFACTOR:=-1 ELSE                        30280525
                IF C="+" THEN SAVEFACTOR:= 0 ELSE                        30280550
                IF C LEQ 9 THEN                                          30280575
                BEGIN                                                    30280600
                    SAVEFACTOR := 10 * C;   NEXTCHAR;                    30280625
                    SAVEFACTOR := SAVEFACTOR + C;                        30280650
                    IF C GTR 9 THEN ERROR (100);                         30280675
                END                                                      30280700
                ELSE                                                     30280720
                BEGIN                                                    30280735
                    ERROR(100);                                          30280750
```

```
                    SAVEFACTOR := 7;                                                    30280765
                END;                                                                    30280780
            END;                                                                        30280800
%                                                                                       30280825
%                                                                                       30280850
%                                                                                       30280875
INTEGER EXPRLEVEL,TX,EXPINVARCNT;%                                                      40018000
    BOOLEAN INBRACKET,INRECORD,SIMPLEVAR;                                               40087000
      SIMPLEVAR := FALSE;                                                               40099000
    CURTYPE := THISID.TYPE;    SIMPLEVAR := TRUE;                                       40104000
      SIMPLEVAR := FALSE;                                                               40109000
          EXPINVARCNT:=EXPINVARCNT+1;%                                                  40120500
          EXPINVARCNT:=EXPINVARCNT-1;%                                                  40121500
    SIMPLEVARIABLE := SIMPLEVAR;                                                        40199500
      IF EXPINVARCNT=0 THEN WRITEEXPR; %                                                40751000
    LABEL EFH;                                                                          50201500
                                        %***  6) OPEN & CLOSE (INPUT) FOR               50204500
                                          % CUMULATIVE FREQUENCY COUNT                  50204550
      BEGIN                                                                             50208100
          GEN("QQJZXL",6,2);                                                            50208200
          INSYMBOL;                                                                     50208300
          GO TO EFH; %                                                                  50208400
      END;                                                                              50208500
EFH:                                                                                    50219500
    EXPRLEVEL := 1;                                                                     60346500
          IF THISID.IDCLASS=VAR OR                                                      60354000
             THISID.IDCLASS=CONST AND BOOLEAN(THISID.FORMAL) THEN %                     60354500
    EXPRLEVEL := 0;                                                                     60383500
PROCEDURE ASSERTSTAT;                                                                   60391100
BEGIN                                                                                   60391200
    GEN("IF NOT(",7,1);                                                                 60391400
    INSYMBOL;   BOOLEXPR;                                                               60391500
    GEN(") THEN",7,2);   GEN("RUNERR(",7,1);   GEN("7,",2,6);                          60391600
    GENINT(CARDCNT);     GEN(";",1,7);                                                  60391700
END OF ASSERTSTAT;                                                                      60391800
          IF CURNAME1="QQQJZXL" THEN FILEHANDLING(6)  ELSE                              60443500
    IF CURSY=ASSERTSY THEN ASSERTSTAT ELSE                                              60457500
          IF PARAM THEN GEN("0",1,7) ELSE BEGIN                                         80129000
                                            GEN("0:",2,6);                              80129100
                                            GENINT(RECSIZE-1);                          80129200
                                        END                                             80129300
          FORWPARAM1[NUMPARAMS] := CURNAME1;                                            80177500
          FORWPARAM2[NUMPARAMS] := CURNAME2;                                            80177600
    INTEGER INDEX, CTYPE, NUMFORWARDS, T, TX, I;                                        80403000
    ALPHA T3;                                                                           80403500
        LABEL LL1; %                                                                    80447010
        LABEL LL2; %                                                                    80496010
        LABEL LL3; %                                                                    80542010
        IF CURLEVEL GEQ MAXTABLES THEN ERROR(101) ELSE                                  80543500
            BLOCKTAB[CURLEVEL+1] := NUMBLOCKS := NUMBLOCKS + 1;                         80543600
              NAMETAB3[CURLEVEL,THISINDEX].FORWARDDEF := 0;                             80553000
              T := NAMETAB3[CURLEVEL,THISINDEX].INFO;                                   80554500
              TX := T + PARAMTAB[T];                                                    80554600
              FOR I:=T+1 STEP 1 UNTIL TX DO                                             80554700
              NEWNAME(FORWPARAM1[I],FORWPARAM2[I],CURLEVEL+1);                          80554800
          REPLACE POINTER(NAMETAB1[CURLEVEL+1,*]) BY 0                                  80637500
                FOR MAXNAMES+1 WORDS;                                                   80637600
```

```
          IF CURLEVEL GEQ LASTPEC THEN ERROR(101); %                                    80643000
%                                                                                       90014100
%                                                                                       90014200
SAVEFACTOR:=0;%    * DEFAULT ZIP IS COMPILE AND GO UNLESS                               90014300
%                  * CHANGED BY THE USE OF THE "S" OPTION                               90014400
%                                                                                       90014500
%                                                                                       90014600
%                                                                                       90042100
% THE FOLLOWING    LINES ADD A "0" ONTO THE FRONT OF THE PROGRAM NAME OR                90042200
% THE FIRST SIX CHARACTERS THEREOF IF IT IS LONGER THAN SIX CHARACTERS                  90042300
% THUS GIVING THE NAME OF THE VALGOL OBJECT CODE FILE PRODUCED.                        90042400
%                                                                                       90042500
    PROGNAME := CURNAME[1:1*35:3*4]; PROGNAMELENGTH := MIN(6,CURLENGTH)+1;             90042600
%                                                                                       90042700
%                                                                                       90042800
BEGIN%                                                                                  90090400
     WRITE(LINE ,NOERRORS);%                                                            90090500
     IF ERR[100]%                                                                       90090600
     THEN WRITE(LINE ,ERROR100MESS);%                                                   90090700
     IF SAVEFACTOR>0 THEN%    *A ZIP IS REQUIRED                                        90090800
$VOIDT                                                                                  90111000
END%                                                                                    90129500
     ("100  ILLEGAL SAVE CONSTANT IN """""S""""" OPTION. THE VALUE 07 IS                91106500
SUBSTITUTED"/"     SO THIS ERROR DOES NOT INCREMENT THE COMPILATION ERRO                91106600
RS COUNT."),%                                                                           91106700
  ("101  PROCEDURES/FUNCTIONS NESTED TOO DEEP."),                                       91106800
    REWIND(XREFFILE);                                                                   92003500
    SORT(PRINTXREF,XREFINPUT,0,XREFMAX,XREFCOMPARE,3,1000,6000);                        92005000
$# PATCH 500 FOR PASCAL.XVI.0 CONTAINS  5 CARDS.  PRT CELLS 25 TO 30
$: THIS PATCH CORRECTS THE DOCUMENTATION FOR THE COMPILERS PRT CELLS 25 TO 27
$: (NOT 21 TO 23). FURTHERMORE, IT USES PRT CELL 30 FOR THE CARD COUNT (IN PLACE
$: OF 27) TO BE CONSISTANT WITH THE OTHER SYSTEM COMPILERS.  PRT CELL 27 IS USED
$: FOR THE PAGE COUNT FORMERLY AT SEQUENCE 10134000.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
INTEGER NUMERRS,          % @R+25: NUMBER OF ERRORS IN PROGRAM.                         10029000
        SAVEFACTOR,       % @R+26: SAVEFACTOR FOR CODE FILE.                            10030000
        PAGECNT,          % @R+27: NUMBER OF PAGES PRINTED.                             10033800
        CARDCNT;          % @R+30: NUMBER OF CARDS READ.                                10034000
  INTEGER  LINECNT, ERRINX; % PAGECNT @ PRT+27                                          10134000
$# PATCH 501 FOR PASCAL.XVI.0 CONTAINS 3 CARDS.  "PRT25" FOR USER-S PASCAL PROG.
$: THIS PATCH INCORPORATES THE PRE-DEFINED IDENTIFIER "PRT25" LOCATED
$: AT PRT CELL 25 AS PER DOCUMENTATION.  (THE DOCUMENTATION MUST BE
$: AMENDED TO DELETE PRT26 AND PRT27 FROM THE PRE-DEFINED IDENTIFIER LIST.)
$: ** NOTE THAT FILE  PASCAL/PRELUDE  MUST BE UPDATED FOR "PRT25".
$: THE VARIABLE "PRT25" MAY BE SET BY THE  Q COMMON = N CONTROL CARD.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
    NEWNAME("50PRT25",0,0);                              %*** "PRT25" ***           20369100
    T3:=INTTYPE;  T3.IDCLASS:=VAR;      % GLOBAL INTEGER VARIABLE                   20369200
    NAMETAB[0,THISINDEX] := T3;                                                     20369300
$# PATCH 502 FOR PASCAL.XVI.0 CONTAINS  3 CARDS.  LINE COUNT WHEN DEBUGGING
$: TO CORRECT THE LINE COUNT WHEN THE DEBUGGING OPTION TO LIST THE ALGOL
$: CODE GENERATED IS SET (*$D+ *), OTHERWISE LINES PER PAGE GOES WRONG.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
DEFINE LINESPERPAGE = 60 #,                                                         10038000
```

```
    IF DUMPOPTION THEN BEGIN IF (LINECNT:=LINECNT+1)≥LINESPERPAGE       20149000
        THEN HEADING; WRITE(LINE,10,ALGOLCARD[*])   END;                20149100
$# PATCH 503 FOR PASCAL.XVI.0 CONTAINS  9 CARDS.  INTEGER TO REAL FOR TYPETAB1
$: WHEN MORE THAN 63 ENTRIES WERE ENTERED IN THE "TYPETAB*" ARRAYS, THE
$: PASCAL COMPILER WAS DISCONTINUED DUE TO INTEGER OVERFLOW, WHICH COULD OCCUR
$: IN A NUMBER OF PROCEDURES AS A RESULT OF ASSIGNING TO AN INTEGER AN ARRAY
$: ELEMENT WHOSE EXPONENT FIELD WAS NOT ZERO.  THE FIELD "ARRTYPE" IS
$: [43:10] AND HAS THE 4 HIGH ORDER BITS IN THE EXPONENT FIELD.  THIS PATCH
$: ALTERS THE DECLARATIONS OF ALL IDENTIFIERS TO WHICH "TYPETAB1" MAY BE
$: ASSIGNED FROM INTEGER TO REAL TO CORRECT THIS ERROR.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
    INTEGER  IT;   REAL  T;                                             50225000
    INTEGER  IT;   REAL  T;                                             50285000
    INTEGER CASETYPE,ADDR,MAXADDR,INDEX,CTYPE,TX,SX, T3,LLIM,ULIM,I;    70244000
    REAL T1, CVAL;                                                      70246000
    INTEGER LEVEL1000, TYP, NAM, NAMTAB, I, J, RECSIZE;                 80020000
    ALPHA  T1, ENAME;                                                   80022000
    INTEGER FIRSTPARAM, CURKIND, P1, PX, I, T3;  REAL T;                80148000
    INTEGER  INDEX, CTYPE, NUMFORWARDS, T3, TX, I;                      80403000
    REAL  T, CVAL;                                                      80404000
$# PATCH 504 FOR PASCAL.XVI.0 CONTAINS 23 CARDS.  IMPLEMENT FORWARD DECLARATIONS
$: FORWARD DECLARATIONS OF PROCEDURES ENDED IN CHAOS DUE TO THE PARAMETERS AND
$: THEIR TYPES NOT BEING KEPT, RESULTING IN GLOBALS BEING REFERENCED WHERE
$: POSSIBLE, AND FORWARD DECLARATIONS OF FUNCTIONS DID NOT WORK AT ALL.
$: THE PROBLEM WAS THAT THE INFORMATION ON THE PARAMETERS WAS BEING STORED
$: IN THE "NAMETAB*" ROWS FOR THE CURRENT LEVEL, WHICH WERE BEING SET TO ZERO
$: ON EXIT FROM PROCEDURE BLOCKS AT THAT LEVEL THEREAFTER.
$: THIS PATCH CORRECTS THE ERROR BY MARKING THE ENTRIES FOR PARAMETERS OF
$: FORWARD PROCEDURES AND FUNCTIONS, SETTING TO ZERO ONLY THOSE ELEMENTS WHICH
$: ARE NOT SO MARKED ON EXIT FROM A BLOCK, AND UNMARKING THE RELEVANT PARAMETERS
$: WHEN THE PROCEDURE OR FUNCTION IS DEFINED.  THE MARKING OF THE PARAMETERS
$: IS DONE IN SUCH A WAY THAT THE SAME IDENTIFIER NAME MAY BE USED AT THE SAME
$: LEVEL WITHOUT SYNTAX ERROR ? TO REPORT THAT THE IDENTIFIER IS ALREADY DEFINED
$: THE UNMARKING REPLACES THE IDENTIFIER NAME IN "NAMETAB*" TO ALLOW FOR THE
$: SAME NAME OR ONE THAT HASHES TO THE SAME PLACE TO HAVE BEEN USED PREVIOUSLY
$: AND NOW DELETED.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
        IF FOUND AND THISID.IDCLASS≥FUNC THEN                           80548000
            NAMETAB3[CURLEVEL,THISINDEX].FORWARDDEF:=0;                 80553000
              (THISID.IDCLASS=FUNC AND NOT FUN)  THEN ERROR(43);        80555100
            TX:=(T:=THISID.INFO)+PARAMTAB[T];  % UNMARK FORWARD PARMS    80556000
            FOR T:=T+1 STEP 1 UNTIL TX DO      %  TO ALLOW REFERENCE     80557000
            BEGIN  T3:=PARAMTAB[I].PARAMNAME;                           80558000
              CURNAME1:=ARS(NAMETAB1[CURLEVEL+1,T3]);                   80559000
              CURNAME2:=    NAMETAB2[CURLEVEL+1,T3];                    80560000
              NAMETAB1[CURLEVEL+1,T3]:=0;                               80561000
              NEWNAME(CURNAME1,CURNAME2,CURLEVEL+1);                    80562000
              IF T3≠THISINDEX THEN BEGIN                                80563000
                 PARAMTAB[I].PARAMNAME:=THISINDEX;                      80564000
                 NAMETAB3[CURLEVEL+1,THISINDEX] :=                      80565000
                 NAMETAB3[CURLEVEL+1,T3];                               80565010
            END  END; % OF UNMARKING FORWARD PARAMETERS.                80566000
        TX:=(T:=NAMETAB3[CURLEVEL,INDEX].INFO)+PARAMTAB[T];             80636100
        FOR T:=T+1 STEP 1 UNTIL TX DO      % MARK FORWARD PARAMETERS     80636200
            NAMETAB1[CURLEVEL+1,PARAMTAB[I].PARAMNAME].[46:1] := 1;     80636210
```

```
          TX:=CURFUNC;  CURFUNC:=IF FUN THEN INDEX ELSE -1;                80645000
          FOR I:=0 STEP 1 UNTIL MAXNAMES DO % LEAVE FORWARD PARAMETERS     80647000
              IF NAMETAB1[CURLEVEL,I]>0 THEN NAMETAB1[CURLEVEL,I]:=0;      80648000
          CURLEVEL:=CURLEVEL-1;  CURFUNC:=TX;                              80649000
    FOR I:=LASTREC STEP 1 UNTIL TOPREC-1 DO % CLEAR RECORD DECS            80693000
$# PATCH 505 FOR PASCAL.XVT.0 CONTAINS  9 CARDS.  CHECK FOR HASH TABLE FULL
$: WHEN THERE ARE "MAXNAMES" IDENTIFIERS AT ONE LEVEL, THE "NAMETAB*" ROWS
$: BECOME FULL AND THIS USED TO PUT THE COMPILER INTO AN INFINITE LOOP,
$: EITHER IN "NEWNAME" OR "SEARCHTAB".  THIS PATCH INSERTS TEST FOR WRAP AROUND
$: LEADING BACK TO THE HASHED STARTING POINT, FOR WHICH IT GIVES SYNTAX ERROR
$: 40, TOO MANY IDENTIFIERS DECLARED.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
DEFINE  HASH(HASH1) = ENTIER((HASH1) MOD MAXNAMES) #;                      20202000
BEGIN  ALPHA TNAME;  INTEGER WRAPAROUND;                                   20209000
      WRAPAROUND:=THISINDEX:=HASH(CURNAME1);                               20210000
        IF THISINDEX=WRAPAROUND THEN TNAME:=0;  % TABLE IS FULL            20216100
    ALPHA TNAME;  INTEGER WRAPAROUND;                                      20237100
      WRAPAROUND:=THISINDEX:=HASH(NAME1);                                  20238000
        IF THISINDEX=WRAPAROUND THEN   % TABLE AT THIS LEVEL IS FULL       20244100
          BEGIN ERROR(40); NAME1:=TNAME; NAME2:=NAMETAB2[TAB,THISINDEX];   20244200
          END;                                                             20244300
$# PATCH 506 FOR PASCAL.XVT.0 CONTAINS  2 CARDS.  RESERVED WORD ENDING AT CC 80
$: IF A RESERVED WORD ENDED AT CARD COLUMN 79 OR 80 AND IF THE "BOLDFACE" FOR
$: RESERVED WORDS OPTION IS SET (*$R+ *), AN INVALID INDEX OCCURRED IN THE
$: SCANNER "INSYMBOL".  THE PROBLEM IS CURED BY CORRECTLY COMPUTING THE STARTING
$: AND ENDING POINT OF THE RESERVED WORDS.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
      BEGIN T1 := CARDLENGTH-CHARCNT-CURLENGTH-1;                          30178000
                FOR CURLENGTH+REAL(CHARCNT=0);                             30181000
$# PATCH 507 FOR PASCAL.XVT.0 CONTAINS  5 CARDS.  "VARIABLE", "SIMPLEVARIABLE"
$: IN PROCEDURE "VARIABLE", "SIMPLEVARIABLE" IS SET TRUE IF A SUBSCRIPT IS
$: SIMPLE, RESULTING IN ALGOL CODE BEING WRITTEN PREMATURELY DURING RECURSIVE
$: CALLS ON PROCEDURE "EXPRESSION", WHICH IN SOME CASES LEAD TO ALGOL SYNTAX
$: ERRORS.  SINCE WRITING THE ALGOL CODE IS DEPENDANT ON "EXPRLEVEL" BEING ZERO,
$: THIS PATCH BUMPS ITS VALUE PRIOR TO ANALYSING THE SUBSCRIPT, AND SETS
$: "SIMPLEVARIABLE" FALSE AFTERWARDS.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
          EXPRLEVEL := EXPRLEVEL+1;  % DO NOT "WRITEEXPR" YET              40120900
          EXPRLEVEL := EXPRLEVEL-1;                                        40121100
          SIMPLEVARIABLE := FALSE;   % RECURSION ON "VARIABLE"             40121200
            EXPRLEVEL := EXPRLEVEL+1;                                      60063900
            EXPRLEVEL := EXPRLEVEL-1;                                      60065100
$# PATCH 509 FOR PASCAL.XVT.0 CONTAINS  1 CARD.  "CONCAT" A FUNCTION OF ANY TYPE
$: THE INTRINSIC FUNCTION "CONCAT" COULD ONLY BE ASSIGNED TO A VARIABLE DECLARED
$: "REAL" TO AVOID TYPE CONFLICT SYNTAX ERRORS.  THIS PATCH MAKES "CONCAT"
$: TYPELESS.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
    CURTYPE := 0;  % ALFATYPE OR REALTYPE                                  50050000
$# PATCH 511 FOR PASCAL.XVT.0 CONTAINS  7 CARDS.  ALLOW UP-LEVEL ADDRESSING
$: TO ALLOW UP-LEVEL IDENTIFIER REFERENCES.  FORMERLY, REFERENCES TO GLOBAL
$: IDENTIFIERS WHICH WERE NOT IN THE OUTER BLOCK WERE FLAGGED BY SYNTAX ERROR
$: 5, UP-LEVEL ADDRESSING NOT IMPLEMENTED DUE TO HARDWARE RESTRICTION.
$: ALTHOUGH THE RESTRICTION EXISTS IN EXTENDED ALGOL, IT IS NOT TRUE THAT THE
```

```
$: RESTRICTION IS DUE TO HARDWARE, FOR UP-LEVEL ADDRESSING IS ALLOWED IN
$: COMPATIBLE ALGOL WITH THE CAUTION THAT IT IS INEFFICIENT (THE IMPLEMENTATION
$: IS SIMILAR TO AN ARRAY ELEMENT REFERENCE).
$: THIS PATCH PERMITS SUCH GLOBAL REFERENCES, EXCEPT
$: (1) THAT IF THE CONTROL VARIABLE OF A FOR STATEMENT IS NOT LOCAL OR IN THE
$:     OUTER BLOCK (PRT) A WARNING IS ISSUED (IN THE FORM OF A SYNTAX ERROR,
$:     BUT THE ERROR COUNT IS NOT INCREMENTED), AND
$: (2) THE RESTRICTION IS STILL APPLIED TO FUNCTION NAMES.  THE MESSAGE FOR
$:     SYNTAX ERROR IS AMENDED ACCORDINGLY.
$:  NILS A OTTE,  UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
   IF ERRNUM<0 THEN ERRNUM:=ABS(ERRNUM) ELSE                          20180900
$  %IF THISLEVEL>1 AND THISLEVEL<CURLEVEL THEN ERROR5;                40103000
$           %IF THISLEVEL>1 AND THISLEVEL#CURLEVEL THEN ERROR5;       50244000
$             %IF THISLEVEL>1 AND THISLEVEL#CURLEVEL THEN ERROR5;     50306000
       IF THISLEVEL#CURLEVEL-1 OR THISINDEX#CURFUNC THEN ERROR(5);     60091000
         IF THISLEVEL>1 AND THISLEVEL<CURLEVEL THEN ERROR(-5);         60276000
     (" 5  FUNCTION NAME NOT ACCESSIBLE AT THIS LEVEL."),              91009000
$# PATCH 512 FOR PASCAL.XVI.0 CONTAINS 24 CARDS. IMPLEMENT STRUCTURED ASSIGNMENT
$: TO ALLOW STRUCTURED ASSIGNMENT.  FORMERLY, AN ASSIGNMENT OF A STRUCTURE
$: WAS NOT IMPLEMENTED, EG  A, B: RECORD ... END;  A := B;.
$: THIS PATCH ATTEMPTS TO IMPLEMENT ASSIGNMENT OF STRUCTURES OF ANY KIND,
$: BUT IT WOULD FAIL IF THE STRUCTURE WERE TRANSLATED INTO A MULTI-DIMENSIONAL
$: ALGOL ARRAY.    THIS IMPLEMENTATION DOES WORK FOR ARRAYS, RECORDS,
$: SUBSTRUCTURES, AND FOR STRUCTURES ALLOCATED IN THE HEAP.
$: THIS PATCH NEEDS PATCH 507 IN ORDER TO SUCCEED.
$: FOR THIS IMPLEMENTATION, THE FOLLOWING DEFINE IS REQUIRED IN THE FILE
$: PASCAL/PRELUDE:  DEFINE ASSIGN(DEST, SOURCE, LENGTH) = REPLACE POINTER(DEST)
$: BY POINTER(SOURCE) FOR LENGTH WORDS #;
$:  NILS A OTTE,  UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
                                                                      60020000
   PROCEDURE WRITESEXPR;    %*** FIX STRUCTURE FOR ASSIGNMENT          60021000
   BEGIN                    %    USED ONLY IN ASSIGNMENT OF STRUCTURES 60022000
     IF INSIDEBRACKETS THEN  IF SYMTAB[NUMSYMS] = "100000,"            60023000
        THEN SYMTAB[NUMSYMS] := ", 0 ] "  ELSE PUTSYM("]");            60024000
     WHILE NUMPOINTERS>0 DO                                           60025000
        BEGIN  NUMPOINTERS := NUMPOINTERS-1;                          60026000
        IF NUMSYMS+4 ≥ MAXSYMS THEN  WRITEEXPR;                       60027000
        REPLACE POINTER(SYMTAB[NUMSYMS+1]) BY                         60028000
             "00-1)DIV00 1022.00 T MOD00 10221";                      60029000
        NUMSYMS := NUMSYMS+4;                                         60030000
        END; % OF WHILE                                               60031000
     WRITEEXPR;  GEN( ",", 1,7 );                                     60032000
   END  WRITESEXPR;                                                   60033000
                                                                      60034000
        %ERROR(95);  % STRUCTURED ASSIGNMENT NOT IMPLEMENTED.         60063000
        GEN("ASSIGN(",7,1);  WRITESEXPR;                              60064000
        EXPRESSION;          WRITESEXPR;                              60065000
        GENINT(TYPETAB1[LEFTTYPE].SIZE);          GEN(")",1,7);       60066000
        IF TYPETAB1[LEFTTYPE].SIZE#TYPETAB1[CURTYPE].SIZE             60067000
           THEN ERROR(95);                                            60068000
      END;                                                            60087000
      CHECKTYPES( LEFTTYPE, CURTYPE );                                60088000
   (" 95  SIZE OF STRUCTURES IN ASSIGNMENT ARE NOT THE SAME."),       91102000
$# PATCH 513 FOR PASCAL.XVI.0 CONTAINS 16 CARDS.  FIX POINTERS VIA POINTERS
$: TO CORRECT THE CODE GENERATED FOR CHAINED REFERENCES THROUGH THE HEAP.
```

```
$: IF FOR POINTERS TO POINTERS!  THE OFFSET FOR COMPONENTS WITHIN RECORDS
$: WAS INCORRECTLY BEING ADDED TO THE RECORD IN THE LEFTMOST REFERENCE, IE THE
$: INNERMOST, INSTEAD OF AT THE EXPECTED LEVEL.
$: FOR EXAMPLE, THE FOLLOWING TWO REFERENCES WOULD BOTH BE TRANSLATED TO
$: THE SAME ALGOL CODE EQUIVALENT TO  HEAP[HEAP[ID+IPART+ICOMP]]:
$:  TDB.PARTB.COMP,    TDB.PART.COMPB,
$: WHEN THE FIRST SHOULD HAVE BEEN:   HEAP[HEAP[ID+IPART]+ICOMP].
$: IN ADDITION, THIS PATCH IMPROVES THE COMPILERS CODE FOR GENERATING THE
$: "MEM" ARRAY SUBSCRIPT.
$:  NILS A OTTE,  UNIVERSITY OF NATAL, DURBAN.   1977-11-14
$:
            IF NUMSYMS+6 ≤ MAXSYMS THEN                                    40175000
              NUMSYMS := NUMSYMS+2;                                        40180400
              IF NUMPOINTERS > 0       % POINTER VIA POINTER              40180500
              THEN BEGIN  REPLACE POINTER(SYMTAB[NUMSYMS+1]) BY            40180600
                   "00-1)DIV00 1022,00 T MOD00 1022]";                     40180700
                   NUMSYMS := NUMSYMS+4;                                   40180800
                   END                                                     40180900
              ELSE  NUMPOINTERS := 1;                                      40181000
      % INBRACKET := FALSE;                                                40191100
        BEGIN NUMPOINTERS := NUMPOINTERS-1;                                40193000
              IF NUMSYMS+4 < MAXSYMS                                       40194000
              THEN BEGIN  REPLACE POINTER(SYMTAB[NUMSYMS+1]) BY            40194100
                   "00-1)DIV00 1022,00 T MOD00 1022]";                     40194200
                   NUMSYMS := NUMSYMS+4;                                   40194300
                   END                                                     40194400
              ELSE ERROR(63); % EXPRESSION IS TOO LONG FOR SYMTAB[*]       40195000
$# PATCH 514 FOR PASCAL.XVI.0 CONTAINS 2 CARDS.  PROCESS TIME FUNCTION FOR RUN
$: PATCH TO CHANGE THE NAME OF THE FUNCTION ON THE B5700 VERSION WHICH SUPPLIES
$: THE PROCESS TIME USED BY THE PASCAL PROGRAM ON THE CURRENT RUN FROM "ELAPSED"
$: WHICH MEANS PLATFORM TIME, TO "CPUTIME" WHICH IS THE WIDELY ACCEPTED TERM
$: FOR THIS QUANTITY.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
    NEWNAME("7CPUTIM","E",0); NAMETAB3[0,THISINDEX]:=T3;                   20390000
          IF CURNAME1="7CPUTIM" AND CURNAME2="E" THEN   % "CPUTIME"        40452000
$# PATCH 516 FOR PASCAL.XVI.0. CONTAINS 2 CARDS. CORRECT "NO LISTING" ERROR
$: THIS PATCH CORRECTS AN ERROR WHEREBY IF LISTING WAS TURNED OFF
$: AND PAGE THROW WAS INVOKED, A HEADING WAS  PRINTED REGARDLESS.
$: DAVID A COOPER, HERIOT-WATT UNIVERSITY.....JUNE, 1978.
$:
             IF CX="L" THEN IF C=1 THEN                                    30264000
               IF LISTOPTION THEN HEADING ELSE                             30264500
$# PATCH 517 FOR PASCAL.XVI.0. CONTAINS 2 CARD.
$: THIS PATCH CORRECTS AN ERROR THAT CAUSED A FILE DECLARATION
$: TO HAVE ITS NAME STRING SPLIT OVER TWO LINES IN THE GENERATED XALGOL.
$:   ALSO CHANGES SYMTAB FORM TYPE REAL TO TYPE ALPHA.
$: DAVID A COOPER, HERIOT-WATT UNIVERSITY.....JUNE, 1978.
$:
ALPHA ARRAY SYMTAB[0:MAXSYMS];     % USED BY "EXPRESSION".                10144000
          IF ALGOLCNT LSS 1 THEN WRITEALGOL;                               80103000
$# PATCH 518 FOR PASCAL.XVI.0. CONTAINS 224 CARDS.
$: THIS PATCH CHANGES THE WAY THAT MULTI-DIMENSION ARRAYS
$: REPRESENTING RECORDS ARE DECLARED. PREVIOSLY THEY WRE DECLARED
$: THE WRONG WAY ROUND FOR XALGOL. THIS PATCH SORTS THE DIMENSIONS
$: INTO ASCENDING ORDER FROM LEFT TO RIGHT AND GENERATES APPROPRIATE
$: DEFINES AND CODE FOR HANDLING THE ARRAYS.
```

```
$; STUART ANDERSON, COMPUTER SCIENCE, HERIOT-WATT UNIVERSITY, JUNE.....1978.
$;
     DEFINE                                                                  10156200
          PERMSUB = 0 #, MAXTOTALSUBSCRS = 100#,                             10156300
          ARRNAM  = 1 #;                                                     10156400
     ARRAY  ARRSUBPERMTAB[0:1,0:MAXTOTALSUBSCRS];                            10156500
     INTEGER   PASSPERMTAB, MAXPERMTAB, REMEMBERPOSN;                        10156600
$                                                                            40080000
     BOOLEAN SIMPLEVARTABLE,INSIDEBRACKETS,INSIDEPARENS;                     40080100
$                                                                            40105000
     IF INSIDEPARENS AND TYPETAB[CURTYPE].STRUCT > 0  AND                    40105100
                        TYPETAB[CURTYPE].FORM < FILES THEN                   40105200
        PUTID("H",1000*THISLEVEL+THISINDEX,5)                                40105300
     ELSE                                                                    40105400
        PUTID("V",1000*THISLEVEL+THISINDEX,5);                               40105500
     INSIDEPARENS := TRUE;                                                   40258100
     INSIDEPARENS := FALSE;                                                  40259100
$                                                                            50243000
               GENID("H",1000*THISLEVEL+THISINDEX,5);                        50243100
$                                                                            50307000
               GENID("H",1000*THISLEVEL+THISINDEX,5);                        50307100
$SET VOIDT                                                                   80052000
$POP VOIDT                                                                   80064000
     DEFINE                                                                  80064005
           LOWSUBS = 0 #,                                                    80064010
           HISUBS  = 1 #,                                                    80064015
           NEXTSUBS= 2 #,                                                    80064020
           MAXNOOFSUBSCRIPTS = 20 #,                                         80064025
           STOPPERSUBTAB = 21 #;                                             80064030
     ARRAY  ARRSUBSCRIPTRANGE[0:2,0:MAXNOOFSUBSCRIPTS];                      80064035
     INTEGER  FIRSTRANGE, NEXTFREEENTRY, PASSSUBRANGE, PREVPASS,             80064040
           MP, POSNO, SUBDIFF;                                               80064045
     IF  ARRAYVAR  THEN  GEN(":",1,7) ELSE  ARRAYVAR := TRUE;                80064050
     IF  NOT PARAM THEN                                                      80064055
     BEGIN                                                                   80064060
         GEN("DEFINE",7,2);                                                  80064065
         GENID("V",LEVEL1000+NAM,5);                                         80064070
         GEN("[",1,7);                                                       80064075
     END;                                                                    80064080
     FIRSTRANGE := STOPPERSUBTAB; NEXTFREEENTRY := 0;                        80064085
     POSNO := 1;                                                             80064090
     MP := 10; FIRSTDIM := TRUE;                                             80064095
     DO                                                                      80064100
     BEGIN                                                                   80064105
         IF FIRSTDIM THEN FIRSTDIM := FALSE ELSE                             80064110
         BEGIN                                                               80064111
         IF NOT PARAM THEN GEN(",",",",1,7);                                 80064112
         END;                                                                80064113
         IF NOT PARAM THEN  GENID("V",(LEVEL1000+NAM)*MP+POSNO,IF MP=10      80064115
         THEN 6 ELSE 7);  POSNO := POSNO + 1;                                80064120
         IF  POSNO = MP THEN  MP := MP*10;                                   80064125
         IF  NEXTFREEENTRY = STOPPERSUBTAB THEN                              80064130
         BEGIN                                                               80064135
             ERROR(0);                                                       80064140
         END                                                                 80064145
         ELSE                                                                80064150
         BEGIN                                                               80064155
```

```
     ARRSUBSCRIPTRANGE[LOWSUBS,NEXTFREEENTRY]:=TYPETAB2[TYP];   80064160
    ARRSUBSCRIPTRANGE[HISUBS,NEXTFREEENTRY] := TYPETAB3[TYP];   80064165
END;                                                            80064170
SUBDIFF := TYPETAB3[TYP] - TYPETAB2[TYP];                       80064175
IF FIRSTRANGE = STOPPERSUBTAB THEN                              80064180
BEGIN                                                           80064185
    FIRSTRANGE := NEXTFREEENTRY;                                80064190
    NEXTFREEENTRY := NEXTFREEENTRY + 1;                         80064195
    ARRSUBSCRIPTRANGE[NEXTSUBS,FIRSTRANGE] := STOPPERSUBTAB;    80064200
END                                                             80064205
ELSE                                                            80064210
BEGIN                                                           80064215
    PASSSUBRANGE := FIRSTRANGE;                                 80064220
    PREVPASS :=  STOPPERSUBTAB; NEXTFREEENTRY:=NEXTFREEENTRY+1;80064225
 WHILE(SUBDIFF ≥  ARRSUBSCRIPTRANGE[HISUBS,PASSSUBRANGE]       80064230
                 -ARRSUBSCRIPTRANGE[LOWSUBS,PASSSUBRANGE]) AND 80064235
         (ARRSUBSCRIPTRANGE[NEXTSUBS,PASSSUBRANGE] ≠            80064240
                                  STOPPERSUBTAB) DO             80064245
        BEGIN                                                   80064250
            PREVPASS := PASSSUBRANGE;                           80064255
            PASSSUBRANGE := ARRSUBSCRIPTRANGE[NEXTSUBS,         8006426
                                              PASSSUBRANGE];    80064265
        END;                                                    80064270
    IF  PREVPASS = STOPPERSUBTAB  THEN                          80064275
    BEGIN                                                       80064280
        IF  SUBDIFF ≥ ARRSUBSCRIPTRANGE[HISUBS,                 80064285
                                        PASSSUBRANGE] -         80064290
                      ARRSUBSCRIPTRANGE[LOWSUBS,                80064295
                                        PASSSUBRANGE] THEN      80064300
        BEGIN                                                   80064305
            ARRSUBSCRIPTRANGE[NEXTSUBS,PASSSUBRANGE] :=         80064310
                          NEXTFREEENTRY - 1;                    80064315
            ARRSUBSCRIPTRANGE[NEXTSUBS,NEXTFREEENTRY-1] :=      80064320
                          STOPPERSUBTAB;                        80064325
        END                                                     80064330
        ELSE                                                    80064335
        BEGIN                                                   80064340
            ARRSUBSCRIPTRANGE[NEXTSUBS,NEXTFREEENTRY-1] :=      80064345
                          FIRSTRANGE;                           80064350
            FIRSTRANGE := NEXTFREEENTRY-1;                      80064355
        END                                                     80064360
END                                                             80064365
ELSE                                                            80064370
BEGIN                                                           80064375
    IF  SUBDIFF ≥ ARRSUBSCRIPTRANGE[HISUBS,PASSSUBRANGE] -      80064380
                  ARRSUBSCRIPTRANGE[LOWSUBS,PASSSUBRANGE]       80064385
                  THEN                                          80064390
    BEGIN                                                       80064395
       ARRSUBSCRIPTRANGE[NEXTSUBS,PASSSUBRANGE] :=              80064400
                     NEXTFREEENTRY - 1;                         80064405
       ARRSUBSCRIPTRANGE[NEXTSUBS,NEXTFREEENTRY-1] :=           80064410
                     STOPPERSUBTAB;                             80064415
    END                                                         80064420
    ELSE                                                        80064425
    BEGIN                                                       80064430
       ARRSUBSCRIPTRANGE[NEXTSUBS,PREVPASS] :=                  80064435
                     NEXTFREEENTRY -1;                          80064440
```

```
                ARRSUBSCRIPTRANGE[NEXTSUBS,NEXTFREEENTRY-1] :=          80064445
                                PASSSUBRANGE;                           80064450
            END                                                         80064455
        END                                                             80064460
    END;TYP:=IF T1.FORM = ARRAYS THEN T1.ARRTYPE ELSE REALTYPE;         80064465
    T1 := TYPETAB[TYP];                                                 80064470
END  UNTIL T1.STRUCT = 0 ;                                              80064475
IF  NOT PARAM  THEN                                                     80064480
BEGIN                                                                   80064485
    GEN("]=",2,6);                                                      80064490
    GENID("H",LEVEL1000+NAM,5);                                         80064495
    GEN("[",1,7);                                                       80064500
    PASSSUBRANGE:= FIRSTRANGE; FIRSTDIM := TRUE;                        80064505
    WHILE  PASSSUBRANGE # STOPPERSUBTAB DO                              80064510
    BEGIN                                                               80064515
        IF FIRSTDIM  THEN  FIRSTDIM := FALSE ELSE GEN(",",1,7);         80064520
        GENID("V",(LEVEL1000+NAM)*(IF PASSSUBRANGE>9 THEN 100 ELSE      8006453
            10)+PASSSUBRANGE+1,IF PASSSUBRANGE>9 THEN 7 ELSE 6);        80064535
        PASSSUBRANGE := ARRSUBSCRIPTRANGE[NEXTSUBS,PASSSUBRANGE];       80064540
    END;                                                                80064545
    GEN("]#;",3,5);                                                     80064550
END;                                                                    80064555
PASSSUBRANGE := FIRSTRANGE;                                             80064560
FIRSTDIM := TRUE; GEN("ARRAY",6,3);  GENID("H",LEVEL1000+NAM,5);        80064565
    GEN("[",1,7);                                                       80064570
WHILE PASSSUBRANGE # STOPPERSUBTAB DO                                   80064575
BEGIN                                                                   80064580
    IF  MAXPERMTAB  LEQ  MAXTOTALSUBSCRS AND PARAM  THEN                80064585
    BEGIN                                                               80064590
        ARRSUBPERMTAB[ARRNAM,MAXPERMTAB] :=                             80064595
                IF  FIRSTDIM  THEN NAM ELSE -1;                         80064600
        ARRSUBPERMTAB[PERMSUB,MAXPERMTAB] := PASSSUBRANGE;              80064605
        MAXPERMTAB := MAXPERMTAB + 1;                                   80064610
    END                                                                 80064615
    ELSE                                                                80064620
    BEGIN                                                               80064625
        IF MAXPERMTAB > MAXTOTALSUBSCRS THEN ERROR(0);                  80064630
    END;                                                                80064640
    IF FIRSTDIM  THEN FIRSTDIM := FALSE  ELSE  GEN(",",1,7);            80064645
    GENINT(ARRSUBSCRIPTRANGE[LOWSUBS,PASSSUBRANGE]);                    80064650
    IF NOT PARAM THEN                                                   80064655
    BEGIN                                                               80064660
        GEN(":",1,7);                                                   80064665
        GENINT(ARRSUBSCRIPTRANGE[HISUBS,PASSSUBRANGE]);                 80064670
    END;                                                                80064675
    PASSSUBRANGE := ARRSUBSCRIPTRANGE[NEXTSUBS,PASSSUBRANGE];           80064680
END;                                                                    80064685
GEN("]",1,7);                                                           80064950
$                                                                       80421000
IF CURLEVEL > 1 THEN                                                    80421010
BEGIN                                                                   80421020
    INTEGER  NAMOFTHING,DIFF;                                           80421030
    BOOLEAN  FIRSTTIME;                                                 80421040
    GEN("BEGIN",6,3);                                                   80421050
    IF  MAXPERMTAB > 0 THEN                                             80421060
    BEGIN                                                               80421070
        PASSPERMTAB := 0;                                               80421080
```

```
    DO                                                                  80421090
    BEGIN                                                               80421100
        REMEMBERPOSN := PASSPERMTAB;                                    80421110
        GEN("DEFINE",7,2);                                              80421120
        NAMOFTHING := ARRSUBPERMTAB[ARRNAM,PASSPERMTAB];                80421130
GENID("V",1000*CURLEVEL+NAMOFTHING,5);                                  80421140
        GEN("[",1,7);                                                   80421150
        FIRSTTIME := TRUE;                                              80421160
        DO                                                              80421170
        BEGIN                                                           80421180
            IF FIRSTTIME THEN FIRSTTIME := FALSE ELSE GEN(",",1         80421190
                                                          ,7);          80421200
            DIFF := PASSPERMTAB-REMEMBERPOSN+1;                         80421210
GENID("V",(1000*CURLEVEL+NAMOFTHING)*(IF DIFF>9 THEN 100 ELSE           80421220
          10)+DIFF,(IF DIFF > 9 THEN 7 ELSE 6));                        80421230
            PASSPERMTAB := PASSPERMTAB + 1; END                         80421270
        UNTIL PASSPERMTAB = MAXPERMTAB OR                               80421280
              ARRSUBPERMTAB[ARRNAM,PASSPERMTAB] # -1;                   80421290
      GEN("]",1,7);                                                     80421300
      GEN("=",1,7);                                                     80421310
      GENID("H",1000*CURLEVEL+NAMOFTHING,5);                            80421320
      GEN("(",1,7);                                                     80421340
      PASSPERMTAB := REMEMBERPOSN; FIRSTTIME := TRUE;                   80421350
      DO                                                                80421360
      BEGIN                                                             80421370
          IF FIRSTTIME THEN FIRSTTIME := FALSE  ELSE GEN(",",           80421380
                                                        1,7);           80421390
          DIFF := ARRSUBPERMTAB[PERMSUB,PASSPERMTAB]+1;                 80421400
          GENID("V",(1000*CURLEVEL+NAMOFTHING)*(IF DIFF>9 THEN          80421410
                100 ELSE 10)+DIFF,(IF DIFF>9 THEN 7 ELSE 6));           80421420
              PASSPERMTAB := PASSPERMTAB +1;                            80421430
          END                                                           80421440
          UNTIL  PASSPERMTAB = MAXPERMTAB  OR                           80421450
                 ARRSUBPERMTAB[ARRNAM,PASSPERMTAB] # -1;                80421460
          GEN("]#;",3,5);                                               80421470
        END                                                             80421480
        UNTIL  PASSPERMTAB = MAXPERMTAB;                                80421490
        MAXPERMTAB := 0;                                                80421500
      END                                                               80421510
    END;                                                                80421520
$                                                                       80608000
    BEGIN                                                               80608010
      BEGIN                                                             80608020
        INTEGER NAM,T1,SCRATCH;                                         80608030
        NAM := PARAMTAB[I].[9:10];                                      80608040
        SCRATCH := NAMETAB[CURLEVEL+1,NAM];                             80608050
        SCRATCH := SCRATCH.TYPE;                                        80608060
        T1 := TYPETAB[SCRATCH];                                         80608070
        IF T1.STRUCT # 0 AND T1.FORM  < FILES  THEN                     80608080
          GENID("H",1000*(CURLEVEL+1)+NAM,5)                            80608090
        ELSE                                                            80608100
          GENID("V",1000*(CURLEVEL+1)+NAM,5);                           80608110
      END;                                                              80608120
    MAXPERMTAB := 0;                                                    90070100
    INSIDEPARENS := FALSE;                                              90070200
$# PATCH 519 FOR PASCAL XVI.0. CONTAINS 1 CARDS. INCREASE RUNTIME STACK.
$!
```

```
      " XALGOL STACK = 2048; STACK = 1024; END."; %                          90120500
$# PATCH 600 FOR PASCAL.XVI.0. CONTAINS 22  CARDS. DAGS DEC77 PATCHES.
$: PATCHES RECEIVED FROM D.IANSMYHR AND TRANSPOSED FROM COSY FORMAT BY
$: DAVID A COOPER. FEBRUARY 1978.
$:
      IF(F1 NEQ SET OR RT NEQ EMPTYSET) %                                    20813000
        AND %                                                                20813050
        (F2 NEQ SET OR LT NEQ EMPTYSET) THEN %                               20813100
        IF(F1 NEQ POINTERS OR RT NEQ NILTYPE) %                              20814000
         AND %                                                               20814050
          (F2 NEQ POINTERS OR RT NEQ NILTYPE) THEN %                         20814100
    BEGIN ERROR(63); %                                                       40023000
$                                                                            50059000
            GEN("PREAD(",6,2);   WRITEEXPR;   GEN(",",1,7); %                50079000
$                                                                            50080000
$                                                                            50081000
            GENID("F",FILEID,5); GEN(",",1,7); %                             50082000
            IF F=NUMERIC THEN %                                              50086010
            BEGIN %                                                          50086050
                GEN(",",1,7);   GENINT(TYPETAB2[CURTYPE]); %                 50086100
                GEN(",",1,7);   GENINT(TYPETAB3[CURTYPE]); %                 50086150
            END ELSE GEN(",0.0",4,4); %                                      50086200
$                SET VOIDT                                                   50088000
$                POP VOIDT                                                   50093000
 IF NAMTAB.IDCLASS=FUNC THEN GEN("FUNCTN",7,2) %                             80037000
                        ELSE GEN("PROCEDU",8,1); %                           80038000
 IF FOUND AND (THISID.IDCLASS=PROC OR THISID.IDCLASS=FUNC) THEN              80548000
$#PATCH 601 FOR PASCAL.XVI.0.CONTAINS 147 CARDS. EXTENDF SET MODS.
$: PATCHES RECEIVED FROM D.IANSMYHR AND TRANSPOSED FROM COSY FORMAT BY
$: DAVID A COOPER. FEBRUARY 1978.
$: THIS PATCH MODIFIES THE SET HANDLING ROUTINES TO ALLOW SETS OF 0..93
$: ELEMENTS.
$: NB. THE RUN TIME SYSTEM MUST BE CHANGED ACCORDINGLY.....
$:     --- --- ---- ------ ---- -- ------- ----------------
%                                                                            40052050
%                                                                            40052055
PROCEDURE SPLIT(SPLITINX,WIDTH); %                                           40052100
VALUE SPLITINX, WIDTH; %                                                     40052150
INTEGER SPLITINX, WIDTH ; %                                                  40052200
BEGIN %                                                                      40052250
    INTEGER I; %                                                             40052300
%                                                                            40052350
    IF NUMSYMS+WIDTH LEQ MAXSYMS THEN %                                      40052400
    BEGIN %                                                                  40052450
        FOR I:=NUMSYMS STEP -1 UNTIL SPLITINX DO %                           40052500
        SYMTAB[I+WIDTH] := SYMTAB[I]; %                                      40052550
        FOR I:=1 STEP 1 UNTIL WIDTH DO %                                     40052600
        SYMTAB[SPLITINX+I-1] := "3000000"; %                                 40052650
        NUMSYMS := NUMSYMS + WIDTH; %                                        40052700
    END %                                                                    40052750
    ELSE                                                                     40052800
    BEGIN %                                                                  40052830
        ERROR(63); %                                                         40052860
        NUMSYMS := 1; %                                                      40052890
    END; %                                                                   40052900
END OF SPLIT; %                                                              40052950
%                                                                            40052960
```

```
%                                                                       40052965
    END; %                                                              40188005
IF TYPETAB[CURTYPE].FORM=SET THEN %         *** SET VARIABLES           40188010
BEGIN %                                     *** --- ---------           40188025
    INTEGER THISSYML, I; %                                              40188050
%                                                                       40188075
    SPLIT(STARTSYM,1);   SYMTAB[STARTSYM] := "SLOAD("; %                40188100
    IF SIMPLEVAR THEN %                                                 40188125
    BEGIN %                                                             40188150
        PUTSYM(","); %                                                  40188175
        PUTID("W",1000*THISLEVEL+THISINDEX,5); %                        40188200
    END %                                                               40188225
    ELSE %                                                              40188250
    IF INBRACKET AND NOT INRECORD THEN %                                40188275
    BEGIN %                                                             40188300
        PUTSYM(",");   THISSYML := NUMSYMS; %                          40188325
        PUTCONST(0);   PUTSYM(" ");   PUTSYM(","); %                    40188350
        FOR I:=STARTSYM+1 STEP 1 UNTIL THISSYML DO %                    40188375
        PUTTEXT(SYMTAB[I]); %                                           40188400
        PUTTEXT("  1]  "); %                                            40188425
    END %                                                               40188450
    ELSE %                                                              40188475
    BEGIN %                                                             40188500
        THISSYML := NUMSYMS; %                                          40188525
        IF INBRACKET THEN PUTSYM("]"); %                                40188550
        FOR I:=1 STEP 1 UNTIL NUMPOINTERS DO %                          40188575
        BEGIN %                                                         40188600
            PUTTEXT("-1)DIV");   PUTTEXT(" 1022,"); %                   40188625
            PUTTEXT(" T MOD");   PUTTEXT(" 1022]"); %                   40188650
        END; %                                                          40188675
        PUTSYM(","); %                                                  40188700
        FOR I:=STARTSYM+1 STEP 1 UNTIL THISSYML DO %                    40188725
        PUTTEXT(SYMTAB[I]); %                                           40188775
        PUTTEXT("  +1  "); %                                            40188800
        IF INBRACKET THEN PUTSYM("]"); %                                40188825
        FOR I:=1 STEP 1 UNTIL NUMPOINTERS DO %                          40188850
        BEGIN %                                                         40188875
            PUTTEXT("-1)DIV");   PUTTEXT(" 1022,"); %                   40188900
            PUTTEXT(" T MOD");   PUTTEXT(" 1022]"); %                   40188915
        END; %                                                          40188930
        NUMPOINTERS := 0; %                                             40188945
    END;                                                                40188960
    PUTSYM(",");   PUTCONST(CARDCNT);   PUTSYM(")"); %                 40188975
END OF SET VARIABLES; %                                                 40188990
$                                                                       40198000
            IF TYPETAB[THISVD.TYPE].FORM=SET THEN                       40274200
            BEGIN %                                                     40274220
                GEN(",",1,7); %                                         40274240
                GENID("W",1000*THISLEVEL+THISINDEX,5); %                40274260
            END; %                                                      40274280
    BOOLEAN FIRST, SPLITTED; %                                          40296000
        PUTTEXT("SETBS(");   PUTTEXT("  3,2,");   PUTCONST(CARDCNT); %  40529000
        PUTSYM(")"); %                                                  40529300
        CURTYPE := EMPTYSET;   CURMODE := NUMBER; %                     40529600
         STARTSYM := NUMSYMS + 1; %                                     40533500
          PUTTEXT(" SETB("); %                                          40536000
            PUTSYM(",");   SYMTAB[STARTSYM] := "SETBS("; %             40544000
```

```
          IF SPLITTED THEN PUTSYM(")"); %                                    40551500
          IF CURSY=COMMA THEN %                                              40552000
          BEGIN %                                                            40552200
              SPLIT(STARTSYM,1);   SYMTAB[STARTSYM] := "SUNIO(" %            40552400
              PUTSYM(",")); %                                                40552600
              SPLITTED := TRUE; %                                            40552800
          END; %                                                             40552850
        NEWTYPE;   T1 := SET;   T1.SIZE := 2;   T1.STRUCT := 0; %            40558000
        CURMODE := NUMBER; %                                                 40561000
        IF CURTYPE=BOOLTYPE THEN %                                           40587000
        IF CURSY NEQ ANDSY THEN ERROR(64);                                   40593000
      END ELSE %                                                             40593100
      IF F=SET THEN %                                                        40593200
      BEGIN %                                                                40593300
        IF CURSY=ASTERISK THEN %                                             40593400
        BEGIN %                                                              40593500
            SPLIT(STARTSYM,1);   SYMTAB[STARTSYM] := "SYNTS(" %              40593600
            PUTSYM(",")); %                                                  40593700
        END ELSE ERROR(64); %                                                40593800
        MODE := NUMBER; %                                                    40593900
      IF F=SET THEN PUTSYM(")"); %                                           40608500
        SPLIT(STARTSYM,1); %                                                 40650000
        IF CURSY=PLUS THEN SYMTAB[STARTSYM] := "SUNIO(" ELSE %               40651000
        IF CURSY=MINUS THEN SYMTAB[STARTSYM] := "SDIFF(" ELSE %              40652000
           ERROR(64); %                                                      40653000
        PUTSYM(",");   MODE := NUMBER; %                                     40654000
$                                                                            40655000
      IF F=SET THEN PUTSYM(")"); %                                           40668500
$                                                                            40688000
IF CURSY=EQLSY THEN SYMTAB[STARTSYM] := "SEQUA(" %                           40713000
ELSE %                                                                       40713150
IF CURSY=NEQSY THEN %                                                        40713300
BEGIN %                                                                      40714000
    SPLIT(STARTSYM,1);   SYMTAB[STARTSYM]:= " NOT  "; %                      40714150
    SYMTAB[STARTSYM+1] := "SEQUA("; %                                        40714300
IF TYPETAB1[LEFTTYPE].FORM=SET THEN %                                        60080100
BEGIN %                                                                      60080200
    SYMTAB[1] := "SSTOR(";   NUMSYMS := NUMSYMS - 3; %                      60080300
    EXPRESSION; %                                                            60080400
    PUTSYM(")");   CHECKTYPES(LEFTTYPE,CURTYPE); %                           60080500
    WRITEEXPR; %                                                             60080600
END ELSE                                                                     60080700
  IF TYPETAB2[TX] LSS 0 OR TYPETAB3[TX] GTR 93 THEN ERROR(51);               70210000
T1.SIZE := TSIZE := 2;   TYPETAB1[TYPETNDEX] := T1; %                        70214000
IF T1.FORM=SET THEN %                                                        80046200
BEGIN %                                                                      80046400
    GEN(",",1,7);   GENTD("W",LEVEL1000+NAM,5); %                           80046600
END; %                                                                       80046800
  IF T1.FORM=SET THEN %                                                      80064700
  BEGIN %                                                                    80064750
      GEN(",0",2,6); %                                                       80064800
      IF NOT PARAM THEN GEN("I1",2,6); %                                     80064850
  END; %                                                                     80064900
      BEGIN %                                                                80608105
     IF T1.FORM=SET THEN %                                                   80608111
      BEGIN %                                                                80608113
          GEN(",",1,7); %                                                    80608115
```

```
                GENID("W",1000*(CURLEVEL+1)+NAM,5); %                          80608117
            END;                                                               80608118
          END; %                                                               80608119
          IF TYPETAB[NAMETAB3[CURLEVEL+1,PARAMTAB[I].PARAMNAME].TYPE           80627200
              ].FORM=SET                                                       80627205
          THEN BEGIN %                                                         80627400
              GEN(",",I,7); %                                                  80627600
              GENID("W",1000*(CURLEVEL+1)+PARAMTAB[I].PARAMNAME                80627800
                  ,5); %                                                       80627801
              END;                                                             80627850
$#PATCH 602 FOR PASCAL.XVI.0 CONTAINS 5 CARDS. CORRECT REPRESENTATION OF "NIL".
$: RECEIVED FROM DAG LANGHYMR ON 6/07/78.
$: DAVID A COOPER , HERIOT-WATT UNIVERSITY... JULY 1978.
NILTYPE := 6;              %*** TYPE OF "NIL" ***                            20363000
T1.FORM := POINTERS;    TYPETAB[6] := T1;                                    20364000
EMPTYSET := 7; %                                                             20364500
T1.FORM := SET;     TYPETAB[7] := T1;                                        20365000
NUMTYPES := 7; %                                                             20365500
$# PATCH 603 FOR PASCAL.XVI.0. CONTAINS 6 CARDS. CORRECTION TO PATCH 601
$:   DAVID A COOPER & S D ANDERSON, HERIOT-WATT UNIVERSITY.   1ST AUGUST 1978
$:
    INTEGER  STARTSYM,FIRSTSYM,MODE,TYPE1,F;                                 40618000
    PUTDUMMY; STARTSYM := FIRSTSYM := NUMSYMS;                               40621000
        SPLIT(FIRSTSYM,1);                                                   40650000
        IF CURSY = PLUS THEN SYMTAB[FIRSTSYM] := "SUNION(" ELSE              40651000
        IF CURSY = MINUS THEN SYMTAB[FIRSTSYM] := "SDIFF(" ELSE              40652000
         ERROR(64);                                                          40653000
$# PATCH 615 FOR PASCAL.XVI.0. CONTAINS 7 CARDS.
$                                                                            40105100
$                                                                            40105200
$                                                                            40105300
$                                                                            40105400
    IF INSIDEPARENS AND SIMPLEVAR AND TYPETAB[CURTYPE].STRUCT > 0 AND        40198500
       TYPETAB[CURTYPE].FORM = FILES  THEN SYMTAB[STARTSYM].[35:6] :=        40198600
                                                          "H";               40198700
$# PATCH 700 FOR PASCAL.XVI.0  HAS  179 CARDS.   REDUCE THRASHING BY CODE CHANGE
$: TO IMPROVE RUN TIME EFFICIENCY BY REARRANGING THE THE COMPILERS CODE.
$: THE COMPILER HAD A HIGH OVERLAY I/O TIME AND HIGH ELAPSED TIME IN RELATION
$: TO THE PROCESS TIME, AND OBSERVATION OF THE B5700 CONFIRMED THAT IT WAS
$: THRASHING IN 32K.  THIS PATCH ATTEMPTS TO REDUCE THE CORE REQUIREMENT BY
$: REARRANGING THE SEGMENTATION OF THE CODE.  LARGE SEGMENTS ARE ELIMINATED
$: SO AS TO AVOID PULLING CODE THAT WILL NOT BE EXECUTED INTO CORE AND TO
$: RELEASE CODE SEGMENTS AS SOON AS EXECUTION HAS PASSED.  FOR EXAMPLE, THE
$: CROSS REFERENCE ROUTINES WERE ALL CONTAINED IN THE LARGE OUTER BLOCK CODE
$: SEGMENT WHICH INCLUDED VARIOUS UTILITY ROUTINES.
$: A FEATURE WHICH CONTRIBUTED SIGNIFICANTLY TO LARGE SEGMENTS WAS THE HIGH
$: NUMBER OF "DEFINES" WHICH RESULTED IN SIZEABLE SECTIONS OF CODE BEING
$: GENERATED IN-LINE, SOMETIMES MANY TIMES IN ONE SEGMENT.  THESE "DEFINES"
$: WERE READILY CHANGED INTO PROCEDURES. (A SIDE EFFECT OF VIRTUALLY ELIMINATING
$: DEFINES FOR CODE IS THAT THE "BEND" OPTION NO LONGER RESULTS IN NUMEROUS
$: BLANK LINES REPEATING THE SAME SEQUENCE NUMBER FOR EVERY "END" IN THE NESTED
$: DEFINES.)
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
$                                                                            10167000
$                                                                            10168000
$                                                                            10169000
```

```
  VALUE             NAME1,NAME2,TABLE,DECL;                                   20016000
  REAL              NAME1,NAME2;                                              20017000
  INTEGER                       TABLE;  BOOLEAN DECL;                         20018000
  FORWARD;                                                                    20019000
PROCEDURE  PRINTERRORS;  FORWARD;                                             20020000
PROCEDURE  HEADING;              %*** PRINTS A HEADING AT START OF NEW PAGE.  20026000
BEGIN                            DEFINE NEWSEGMENT = HERE #;                 20027000
END OF HEADING;                                                             20033000
PROCEDURE  PRINTLINE;            %*** PRINTS A PASCAL SOURCE CODE LINE       20036000
BEGIN                            DEFINE NEWSEGMENT = HERE #;                 20037000
END OF PRINTLINE;                                                           20047000
PROCEDURE  NEWCARD;              %*** READS A NEW PASCAL SOURCE CODE CARD    20050000
BEGIN                            DEFINE RESULT = ICARD[*], FTC #;           20051000
    REPLACE XLINEPNT BY " " FOR 16 WORDS;                                     20056000
    REPLACE LINEPNT  BY CARDPNT FOR 10 WORDS, XLINEPNT FOR 6 WORDS;           20057000
END OF NEWCARD;                                                             20061000
DEFINE GEN(GEN1,GEN2,GEN3) = GENI(TRUE,GEN1,GEN3,GEN2) #,                   20063100
GENID(GENID1,GENID2,GENID3)= GENI(FALSE,GENID1,GENID2,GENID3) #;            20063200
                                                                            20063300
PROCEDURE  GENI(GENT, TXT, NUM, N );                                        20063400
VALUE GENT, TXT, NUM, N;                                                    20063500
BOOLEAN GENT;  ALPHA TXT;  INTEGER NUM, N;                                  20063600
BEGIN                            DEFINE START = NUM #,  NDIG = N #;         20063700
                                                                            20063800
    IF GENT THEN                  %*** GENERATE A TEXT "TXT", CONSISTING OF 20064000
    TEXT[0] := TXT;                                                         20067000
END                                                                         20070000
ELSE                             %*** GENERATE AN ALGOL IDENTIFIER.         20073000
    CH[0] := TXT;                                                           20076000
END END  GENI;                                                              20079000
PROCEDURE  GENINT( N );                                                     20082000
VALUE N;  INTEGER N;                                                        20083000
BEGIN                            DEFINE RESULT = ALGOL CODE #;              20084000
    INTEGER NABS, NSIZE;                                                    20085000
END OF GENINT;                                                              20097000
PROCEDURE WRITEALGOL;             %*** WRITES A COMPLETED ALGOL CARD TO     20145000
    DEFINE NEWSEGMENT = HERE #;                                             20146100
    DEFINE NEWSEGMENT = HERE #;                                             20168100
    DEFINE NEWSEGMENT = HERE #;                                             20180100
    DEFINE NEWSEGMENT = HERE #;                                             20193100
ALPHA THISID, CURNAME1, CURNAME2;  % USED IN SCANNER                        20205000
PROCEDURE  SEARCHTAB( TAB );  %*** SEARCH NAME TABLE "TAB" FOR THE          20208000
VALUE TAB;  INTEGER TAB;      %***  IDENTIFIER JUST READ.                   20208100
END OF SEARCHTAB;                                                           20221000
PROCEDURE  SEARCH;            %*** SEARCH ALL TABLES CURRENTLY IN USE.      20223000
BEGIN                         DEFINE RESULT = THISID #;                     20224000
END OF SEARCH;                                                              20233000
PROCEDURE  NEWNAME( NAME1,NAME2, TAB );                                     20236000
VALUE NAME1, NAME2, TAB;                                                    20236100
ALPHA NAME1, NAME2;  INTEGER TAB;                                           20236200
END OF NEWNAME;                                                             20250000
    DEFINE NEWSEGMENT = HERE #;                                             20515100
    DEFINE NEWSEGMENT = HERE #;                                             20533100
    DEFINE NEWSEGMENT = HERE #;                                             20546100
PROCEDURE  CHECKTYPES( LEFTTYPE, RIGHTTYPE );                               20802000
VALUE LEFTTYPE, RIGHTTYPE;  INTEGER LEFTTYPE, RIGHTTYPE;                    20803000
BEGIN                                                                       20804000
```

```
    REAL TT1, TT2;  INTEGER F1, F2, LT, RT;                                  20805000
END OF CHECKTYPES;                                                           20838000
PROCEDURE  FILEPARAM( DEFAULTFILE );   %*** CHECKS THE FIRST PARAMETER      20844000
VALUE DEFAULTFILE; INTEGER DEFAULTFILE;%*** TO SEE IF IT IS A FILE.          20844100
BEGIN              DEFINE RESULTS = FILENAME & LPARFOUND #;                  20845000
END OF FILEPARAM;                                                            20869000
REAL CURVAL;  INTEGER CURLENGTH;                                             20872000
                                                                             20873000
PROCEDURE  CONSTANT( CVAL, CTYPE );                                          20874000
REAL CVAL;  INTEGER CTYPE;                                                   20875000
BEGIN                                                                        20876000
    INTEGER TFORM;  BOOLEAN SIGNED, NEGATIVE;                                20876100
END OF CONSTANT;                                                             20921000
$                                                                            30082000
ALPHA  C, CX;        %( CURNAME1 & CURNAME2 MOVED TO 20205000 )              30083000
INTEGER LASTCHARPOS;  %( CURVAL, CURLENGTH MOVED TO 20872000 )               30084000
PROCEDURE  INSYMBOL;          %*** IDENTIFIES THE NEXT SYMBOL ******         30087000
BEGIN                                                                        30087100
                                                                             30087200
    PROCEDURE  NEXTCHAR;      %*** GETS THE NEXT CHARACTER.                  30088000
    END OF NEXTCHAR;                                                         30093000
$                                                          SET VOIDT         30095000
$                                                          POP VOIDT         30098000
    DEFINE  T1 = EXP #;       % USED AT 30178000                            30099100
            BEGIN                  DEFINE NEWSEGMENT = HERE #;               30261100
            END NEWSEGMENT;                                                  30282200
$                                                                            40016000
$                                                                            40017000
INTEGER  EXPRLEVEL;                                                          40018000
DEFINE PUTSYM(S) = PUTTEXT( (S)&1[41:5:6] ) #;                               40029000
$                                                          SET VOIDT         40029900
$                                                          POP VOIDT         40033000
DEFINE PUTDUMMY = PUTTEXT("3000000") #;                                      40041000
$                                                          SET VOIDT         40042000
$                                                          POP VOIDT         40044000
PROCEDURE  WRITEEXPR;         %*** WRITE GENERATED ALGOL EXPRESSION          40053000
    REAL SX;  INTEGER T1, TX;                                                40054100
END OF WRITEEXPR;                                                            40066000
PROCEDURE  CHECKEXPR( LLIM, ULIM );     %*** WRITE CODE TO CHECK VALUE       40069000
    VALUE LLIM, ULIM;  INTEGER LLIM, ULIM;                                   40069100
BEGIN                                  DEFINE CHECK = VALUE #;               40070000
END OF CHECKEXPR;                                                            40077000
    INTEGER T1, T5;           % USED ONCE EACH                              40086100
      T1:=T.FIRSTWITHSYM;  T5:=T.LASTWITHSYM;                                40094000
      FOR T1:=T1 STEP 1 UNTIL T5 DO PUTTEXT(WITHTAB[T1]);                    40095000
    DEFINE  T1 = T #;    % USED AT 40558000                                  40298000
$                                                          SET VOIDT         40299000
$                                                          POP VOIDT         40309000
                                                                             40331000
        PROCEDURE PARAMETER;  %*** CHECK THAT THE FUNCTION HAS 1 PARAM.      40332000
          BEGIN                                                              40333000
          INSYMBOL;                                                          40334000
          IF CURSY=LPAR                                                      40335000
          THEN BEGIN                                                         40336000
            PUTSYM("(");  INSYMBOL;  EXPRESSION;                             40337000
            IF TYPETAB1[CURTYPE].FORM=NUMERIC THEN CURTYPE:=INTTYPE;         40338000
            IF CURSY#RPAR THEN BEGIN ERROR(3); SKIP(RPAR) END;               40339000
```

```
            PUTSYM(")"); IF CURSY=RPAR THEN INSYMBOL;                    40340000
          END ELSE ERROR(3); % OR ERROR(58)                              40341000
        END OF PARAMETER;                                                40342000
                                                                         40350000
$                                                                        60396000
    BEGIN     LABEL LABFOUND;                                            60399000
          THISID.IDCLASS=CONST AND BOOLEAN(THISID.FORMAL) OR             60423000
          THISID.IDCLASS=FUNC                                            60423200
        THEN  ASSIGNMENT  ELSE                                           60424000
$                                                       SET VOIDT        70013000
$                                                       POP VOIDT        70016000
  VALUE             RECTAB.FIRSTADDR;                                    70018000
  INTEGER           RECTAB.FIRSTADDR.LASTADDR;                           70019000
$                                                       SET VOIDT        70022000
$                                                       POP VOIDT        70034000
                                                                         70035000
PROCEDURE  TYPEDECL( TTYPE, TSIZE );     %***** TYPE DECLARATION *****  70036000
  INTEGER           TTYPE, TSIZE;        %*****************************  70037000
BEGIN                                                                    70038000
    INTEGER RECINX, ARRSTRUCT, TX, SX, T, N;  REAL T1, T2, T3;           70039000
    BOOLEAN  FIRST, PACKED;                                              70040000
                                                                         70041000
$                                                                        70042000
    END TYPERR;                                                          70048000
  PROCEDURE  SUBRANGE;                   %*** SUBRANGE DECLARATION ***   70050000
  BEGIN                                  %*****************************  70051000
    REAL     VALX1,  VALX2, T1;                                          70052000
    INTEGER TYPEX1, TYPEX2;                                              70053000
                                                                         70054000
    CONSTANT(VALX1,TYPEX1);                                              70055000
    IF TYPETAB1[TYPEX1].FORM>CHAR THEN ERROR(11);                        70056000
    IF CURSY#DOUBLEDOT THEN ERROR(53);                                   70057000
    INSYMBOL;                                                            70058000
    CONSTANT(VALX2,TYPEX2);                                              70059000
    IF TYPEX1>0 AND TYPEX2>0 THEN                                        70060000
      IF TYPEX1#TYPEX2 THEN ERROR(11) ELSE                               70061000
      IF VALX1>VALX2   THEN ERROR(54);                                   70062000
    IF (T1:=TYPETAB1[TYPEX1].FORM) = SYMBOLIC THEN  T1:=SUBTYPE;         70063000
    NEWTYPE;  TTYPE:=TYPEINDEX;                                          70064000
    T1.SIZE:=TSIZE:=1;  T1.STRUCT:=0;  T1.MAINTYPE:=TYPEX1;              70065000
    TYPETAB1[TYPEINDEX]:=T1;                                             70066000
    TYPETAB2[TYPEINDEX]:=VALX1;  TYPETAB3[TYPEINDEX]:=VALX2;             70067000
  END OF SUBRANGE;                                                       70068000
                                                                         70069000
                                        DEFINE DEC = POINTER #;          70117100
                                        DEFINE DEC = ARRAY #;            70143100
                                        DEFINE DEC = FILE #;             70180100
                                        DEFINE DEC = SET #;              70200100
                                        DEFINE DEC = RECORD #;           70220100
  LABEL CASEPART, EXIT;                                                  70247000
  BEGIN                                 DEFINE DEC = VARIANT #;          70285100
  LABEL CASETYPEID;                                                      70285200
  END;                                                                   70349100
                                        DEFINE DEC = FILE #;             80066100
            GEN(""/",2,6);                                               80107000
                                        DEFINE DEC = LABEL #;            80424100
                                        DEFINE DEC = CONST #;            80447100
```

```
                                        DEFINE DEC = TYPE #;                80475100
                                        DEFINE DEC = VAR #;                 80496100
    IF    CURSY=FUNCSY OR CURSY=PROCSY    %                                 80540900
    THEN BEGIN                            DEFINE DEC = CODE #;              80540910
    END OF SEGMENT FOR PROCEDURE DECLARATIONS;                              80658100
$# PATCH 701 FOR PASCAL.XVI.0 CONTAINS 14 CARDS.  REDUCE THRASHING BY ARRAY CUTS
$: TO IMPROVE RUN TIME EFFICIENCY BY REDUCING ARRAY SIZES.  THE MOST SIGNIFICANT
$: CONTRIBUTION TO THE COMPILERS THRASHING BEHAVIOUR WAS THE EXCESSIVELY LARGE
$: DATA ARRAYS. THIS PATCH SUCCEEDS IN DRASTICALLY REDUCING THE CORE REQUIREMENT
$: OF THE COMPILER BY MAKING MOST OF THE LARGE ARRAYS MUCH SMALLER WITHOUT
$: IMPOSING UNREASONABLE RESTRICTIONS.  IN PARTICULAR, THE THREE ARRAYS,
$: NAMETAB1, NAMETAB2, NAMETAB3 WERE EACH [0:50, 0:10022], AND HAVE BEEN REDUCED
$: TO [0:30, 0:307].  THESE REDUCTIONS HAVE NOT PREVENTED THE COMPILATION OF
$: A LARGE PASCAL PROGRAM OF ABOUT 4000 LINES, NAMELY THE P4 PASCAL COMPILER
$: FROM ZURICH.  IN FACT, PRIOR TO THE CHANGES INTRODUCED BY PATCHES 700 & 701,
$: THE P4 PASCAL COMPILER TOOK 60 MINUTES ELAPSED TIME TO COMPILE, WHICH WAS
$: REDUCED TO 9 MINUTES BY THESE PATCHES, WHILE THE PROCESS TIME HAS REMAINED
$: CONSTANT AT 9 MINUTES.
$:**** NOTE THAT IF "MAXNAMES" IS CHANGED THEN THERE ARE 7 DEFINES IN THE FILE
$:      PASCAL/PRELUDE THAT MUST ALSO BE CHANGED.
$: "MAXNAMES" IS CHOSEN AS A PRIME NUMBER AS IT IS USED AS A MODULUS FOR A HASH
$: FUNCTION.  THE PASCAL IDENTIFIERS ARE TRANSLATED TO ALGOL NAMES USING LEVEL
$: AND HASH INDEX.  HENCE CHANGING "MAXNAMES" CHANGES THE ALGOL NAMES FOR
$: "INPUT", "OUTPUT", & "PRT25".
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
DEFINE MAXTABLES  = 30 #,     %MAX NUMBER OF LEVELS  IN IDENTIFIER TABLE.10042000
       MAXNAMES   =307 #,     %MAX NAMES IN EACH ROW OF IDENTIFIER TABLE.10043000
                              % ONLY USED IN WITH STATEMENT TO TEST      10044001
       MAXCASES   =64 #,      %MAX LABELS IN A CASE-STATEMENT.          10045000
       MAXLABS    =50 #,      %MAX NUMBER OF LABELS IN PROGRAM.         10046000
       MAXPARAMS  =200 #,     %MAX NUMBER OF PARAMETERS IN WHOLE PROGRAM.10047000
       MAXTYPES   =250 #,     %MAX NUMBER OF DIFFERENT TYPES.           10048000
       MAXCONSTS  =100 #,     %SIZE OF TABLE FOR CONSTANTS.             10049000
       MAXWITHSYMS= 70 #,     %MAX NUMBER OF SYMBOLS USED BY WITH-STATMS.10051000
       MAXSYMS    =200 #,     %MAX NUMBER OF SYMBOLS IN ONE EXPRESSION. 10052000
       LISTLENGTH =100 #,     %MAX LENGTH OF VAR & PARAM LISTS.         10053000
       MAXEXTFILES=10 #,      %MAX NUMBER OF EXTERNAL FILES.            10054000
       MAXFILES   =10 #,      %MAX NUMBER OF FILES DECLARED AT ONE TIME. 10055000
       MAXPNTRS   =10 #;      %MAX NUMBER OF UNDECLARED POINTERS (FORWD).10056000
$# PATCH 702 FOR PASCAL.XVI.0 CONTAINS  4 CARDS.  BOOLEAN ARRAY "ERR" 120 TO 4
$: TO EXTEND THE REDUCTIONS OF PATCH 701 TO THE BOOLEAN ARRAY "ERR" FOR NOTING
$: THE SYNTAX ERRORS THAT HAVE OCCURRED.  THIS PATCH COMPRESSES THE ARRAY FROM
$: 120 WORDS TO 4 WORDS BY USING 32 BITS IN EACH WORD.
$: IN ADDITION, THIS PATCH INSERTS THE ERROR COUNT ON THE LEFT OF THE LINE
$: WHICH REPORTS THE SYNTAX ERRORS.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
ARRAY ERRP[0:3];   % HOLDS 128 BITS     % RECORDS ERROR MESSAGES USED.  10156000
DEFINE  ERR[ERR1] = BOOLEAN(0&ERRP[ERR1.[6:2]][0:ERR1.[4:5]:1]) #;      10156100
    ERRP[ERRNUM.[6:2]]:=ERRP[ERRNUM.[6:2]] & 1[ERRNUM.[4:5]:0:1];       20182000
    REPLACE POINTER(ERRLINE[0])+4 BY NUMERRS FOR 4 DIGITS;              20194900
$# PATCH 703 FOR PASCAL.XVI.0 CONTAINS  6 CARDS.  REDUCE THRASHING BY SAVE CORE
$: TO IMPROVE RUN-TIME EFFICIENCY BY REDUCING NON-OVERLAYABLE AREAS.
$: THIS PATCH REDUCES THE SAVE CORE REQUIREMENTS BY DECREASING THE FILE BLOCK
$: SIZES AND ALSO THE NUMBER OF BUFFERS WITHOUT UNDULY RETARDING THE COMPILATION
$: SPEED.  THE SIZE OF THE DISK AREAS IS KEPT A MULTIPLE OF THE ORIGINAL BLOCK
```

```
$: SIZE WHERE RELEVANT TO AVOID INCOMPATIBILITY PROBLEMS.  COMPARABLE REDUCTIONS
$: IN BLOCK SIZES OF THE OBJECT PROGRAM ARE ALSO MADE.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.     AUG - NOV 1977.
$:
FILE CARD "SOURCE" (1,10,30);       % PASCAL SOURCE CODE INPUT FILE       10035000
FILE PASCALGOL DISK SERIAL [20:300] (1,10,30,SAVE 0);  % ALGOL CODE FILE10037000
FILE XREFFILE DISK SERIAL [20:3000] (1,3,30);    % FOR CROSS REFERENCE  10137000
            IF RECSIZE=1 OR RECSIZE=10 THEN GENINT(30)                    80119000
            GEN(",SAVE",6,3);                                             80122000
            GEN("30);", 4,4);                                             80123000
$# PATCH 704 FOR PASCAL.XVI.0  HAS  8 CARDS.    REDUCE OVERHEADS IN COPYING FILE
$: TO REDUCE THE COMPILER-S OVERHEADS.  FIRSTLY, THE ALGOL CODE FILE
$: PASCRUN/DISK  IS RENAMED  PASCAL/PRELUDE.  ORIGINALLY, THE COMPILER COPIED
$: THE PASCAL/PRELUDE FILE INTO THE GENERATED CODE FILE BEFORE STARTING TO
$: TRANSLATE THE PASCAL PROGRAM.  THIS PATCH SAVES THE 3 SECONDS OR SO REQUIRED
$: FOR THIS BY SETTING THE "TAPE" OPTION FOR THE ALGOL COMPILER AND LABEL
$: EQUATING THE TAPE FILE TO PASCAL/PRELUDE.  THE OVERHEAD TO THE ALGOL COMPILER
$: IS NEGLIGIBLE.  THE ADVANTAGE IS EVEN GREATER IF THE PROGRAM FAILS TO
$: COMPILE SYNTAX FREE.  THE FILE PASCAL/PRELUDE IS NO LONGER REFERENCED
$: DIRECTLY IN THE PASCAL COMPILER.
$: SEE PATCH 711.  THIS NEEDS PATCH 705.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.     AUG - NOV 1977.
$:
      ERRORS   (I5," ERRORS DETECTED ",20("#") /),                        10188000
      ALIST    ("$ SET LIST "),                                           10189000
      MERGE    ("$ SET TAPE  RESET $" /                                   10190100
                "$ RESET TAPE", T73,"99000000" ),                         10190200
      TERMMESS ("****    COMPILATION TERMINATED.");                      10192000
WRITE(PASCALGOL,MERGE);  % ALGOL MUST COMPILE PRELUDE FIRST               90022000
$                                                        SET VOIDT        90023000
$                                                        POP VOIDT        90032000
$:      "; ALGOL FILE TAPE= PASCAL/PRELUDE SERIAL; ALGOL FILE CARD=",     90119000
$# PATCH 705 FOR PASCAL.XVI.0 CONTAINS 21 CARDS.  GENERATE A BETTER ZIP
$: THIS PATCH TIDIES UP THE CODE THAT GENERATES THE ZIP TO PASS CONTROL TO THE
$: COMPATIBLE ALGOL COMPILER.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.     AUG - NOV 1977.
$:
$                                                                         90013000
    PROGNAME := IF CURLENGTH < 7                                          90042000
                THEN "      "&CURNAME1[41:6×CURLENGTH-1:6×CURLENGTH]      90042010
                ELSE CURNAME2.[5:6]&CURNAME1[41:35:36];                   90042020
$:  ARRAY ZIPARRAY[0:16];                                                 90092000
    DEFINE PPROGNAME = 13 #,   PALGOLNAME = 14 #,                         90095000
           PLIBRARY  = 15 #,   PUSER      = 16 #,                         90096000
           P(P1)     = POINTER(ZIPARRAY[P1])+1 FOR 7 #;                   90097000
$                                                        SET VOIDT        90098000
$                                                        POP VOIDT        90104000
$                                                                         90109000
    ZIPARRAY[PPROGNAME]:=PROGNAME;  ZIPARRAY[PALGOLNAME]:=ALGOLNAME;      90112000
    ZIPARRAY[PLIBRARY]:= IF SAVEFACTOR>0 THEN "LIBRARY" ELSE              90113000
                         IF SAVEFACTOR<0 THEN " SYNTAX" ELSE " & RUN ";   90114000
    ZIPARRAY[PUSER]:=USER;                                                90115000
    REPLACE POINTER(ZIPARRAY[*]) BY "CC COMPILE ",                        90116000
         P(PPROGNAME), "/", P(PUSER),                                     90117000
         " XALGOL ", P(PLIBRARY),                                         90118000
         "; ALGOL FILE TAPE= PASCAL/PRELUDE SERIAL; ALGOL FILE CARD=",    90119000
         P(PALGOLNAME), "/", P(PUSER), " SERIAL; END.";                   90120000
```

```
$                                                                SET VOIDT    90121000
$                                                                POP VOIDT    90128000
$# PATCH 708 FOR PASCAL.XVT.0 CONTAINS 25 CARDS.  LINE PRINT FILE MAY BE DISK
$: TO ENABLE THE COMPILER-S PRINT FILE TO BE LABEL EQUATED TO DISK AS FOR OTHER
$: B5700 COMPILERS.  IN PARTICULAR, THIS PATCH CHANGES THE NAME TO LINE TO BE
$: CONSISTENT WITH ALL THE SYSTEM COMPILERS.  THE ABILITY TO LABEL EQUATE FILE
$: "LINE" TO DISK IS NECESSARY IF THE COMPILER IS TO BE USED FROM A TERMINAL.
$: NOTE THAT A BLOCKED FILE SHOULD NOT HAVE VARIABLE LENGTH RECORDS IF IT IS
$: TO BE LABEL EQUATED TO A PRINTER.  IF LESS THAN THE MAX NUMBER OF WORDS PER
$: RECORD IS WRITTEN, THE BALANCE OF THE RECORD REMAINS UNCHANGED FROM WHAT WAS
$: LAST IN THE FILE BUFFER, SO THAT ON BEING PRINTED, "GARBAGE" APPEARS AT THE
$: END OF SUCH LINES.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
SAVE FILE OUT LINE DISK SERIAL [20:1200] (1,17,90,SAVE 1); % PRINT FILE 10036000
                        % AVOID BLOCKING RECORDS OF VARIABLE LENGTH      10036001
ARRAY XCARD, ALGOLCARD[0:9], LINES, XLINE[0:16];                         10130000
                        % AVOID BLOCKING VARIABLE LENGTH RECORDS         10130001
ARRAY HEADTEXT, ERRLINE[0:16];                                           10133000
      WRITE( LINE[NO],17,XLINE[*]);                                      20042000
      WRITE( LINE[NO],17,XLINE[*]);                                      20043000
    WRITE(LINE, 17,LINES[*]);                                            20045000
    WRITE(LINE, 17,ERRLINE[*]);                                          20195000
    LINEPNT:=POINTER(LINES[1]);                                          20315000
    REPLACE LINEPNT=8 BY " " FOR 17 WORDS;                               20317000
    REPLACE XLINEPNT=8 BY LINEPNT=8  FOR 17 WORDS;                       20318000
    REPLACE POINTER(ERRLINE[*]) BY "****    ", LINEPNT FOR 16 WORDS;     20319000
    REPLACE ALGOLPNT BY LINEPNT FOR 9 WORDS;                             20321000
    REPLACE POINTER(HEADTEXT[*]) BY LINEPNT FOR 10 WORDS, "PAGE 1  ",    20326000
                                    LINEPNT FOR  6 WORDS;                20326100
      WRITE(LINE, 17,XREFLINE[*]);                                       20549000
      LOCK( LINE, * ); % & CRUNCH                                        20550000
          WRITE(LINE, 17,XREFLINE[*]);  LINECNT:=LINECNT+1;              20560000
        WRITE(LINE, 17,XREFLINE[*]);  LINECNT:=LINECNT+1;                20571000
          WRITE(LINE, TERMMESS);                                         90084000
      WRITE(LINE, NOERRORS);                                             90111000
    WRITE(LINE, ERRORS,NUMERRS);                                         91110000
    WRITE(LINE, ERRORMESS1[1]);                                          91112000
    WRITE(LINE, ERRORMESS2[1-60]);                                       91114000
$# PATCH 709 FOR PASCAL.XVT.0 CONTAINS 17 CARDS. NO PRINT IF NO LIST & NO ERRORS
$: TO OPEN THE PRINT FILE ONLY IF THE LIST OPTION IS SET OR IF SYNTAX ERRORS
$: ARE DETECTED.  IF THE FIRST CARD IN THE PASCAL SOURCE RESETS THE LIST OPTION
$: (*$L- *) AND NO SYNTAX ERRORS ARE DETECTED, THEN THE PRINT FILE WILL NOT BE
$: CREATED (EVEN FOR THE HEADING) AS FOR OTHER COMPILERS.  IN PARTICULAR, THIS
$: IMPLEMENTATION DOES NOT REQUIRE A TEST PRIOR TO PRINTING EACH LINE TO
$: DETERMINE WHETHER A HEADING HAS BEEN PRINTED.  IT ONLY DOES THIS TEST WHEN
$: THE LIST OPTION IS SET AFTER THE FIRST CARD OR EXPLICITLY THEREAFTER, OR
$: IN THE "PRINTERRORS" ROUTINE.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
    IF PAGECNT=1 THEN WRITE(LINE[NO],17,HEADTEXT[*]) ELSE                20029900
    WRITE( LINE[PAGE]);                                                  20030000
    WRITE( LINE[DBL],17,HEADTEXT[*]);                                    20031000
    IF NOT LISTOPTION THEN                                               20194000
        BEGIN IF PAGECNT=0 THEN HEADING;  PRINTLINE  END;                20194100
    REPLACE POINTER(HEADTEXT[*])+45 BY TEXTPNT+5 FOR 2,"/",              20329000
            TEXTPNT+1 FOR 2, "/", TEXTPNT+3 FOR 2;                       20330000
```

```
    NEWCARD; LISTOPTION:=CHECKOPTION:=TRUE;  % DEFAULT                    20402100
    INSYMBOL;       % ANALYSING FIRST CARD MAY CHANGE DEFAULT LIST OPTN   20402200
    IF LISTOPTION AND PAGECNT=0 THEN HEADING;  % ON FIRST PAGE.           20402300
          IF LISTOPTION THEN IF PAGECNT=0 THEN HEADING; % ON FIRST PAGE30282100
C :=  " ";      % TO INITIALIZE "INSYMBOL"                               90034000
INITIALIZE;    % COMPILER TABLES, NEWCARD, INSYMBOL                      90035000
$                                                                       90036000
IF PAGECNT > 0      % THERE HAS BEEN SOME LISTING                       90088000
THEN BEGIN  WRITE( LINE[DBL] );  WRITE( LINE[DBL] )  END;               90089000
    IF PAGECNT>0 THEN   % THERE HAS BEEN LISTING                        90110000
$# PATCH 710 FOR PASCAL.XVI.0 CONTAINS 4 CARDS.  NO OVERPRINTING WITH BLANK LINE
$: TO PREVENT OVERPRINTING WITH BLANK LINES.  IF THE OPTION FOR "BOLDFACE"
$: PRINTING OF RESERVED WORDS IS SET (*$R+ *) THEN EACH LINE IS CONSTRUCTED BY
$: 2 OVERPRINTS FOR THE RESERVED WORDS ONLY, THEN ONE PRINT OF THE FULL TEXT.
$: THE AIM OF THIS PATCH IS TO SKIP THE OVERPRINTING FOR ALL THOSE LINES IN
$: WHICH NO RESERVED WORDS OCCUR.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
DEFINE  RESWORDPRESENT = RESWORDOPTION.[1:1] #;                         10159100
    IF REAL(RESWORDOPTION) = 3 THEN  % RESERVED WORD IS PRESENT         20040000
    RESWORDOPTION := RESWORDOPTION AND TRUE;   % RESET RESWORDPRESENT   20057100
          RESWORDOPTION := BOOLEAN(3);  % SET RESWORDPRESENT BIT       30179000
$# PATCH 711 FOR PASCAL.XVI.0 CONTAINS 10 CARDS.  PASCOO1/USERCODE UNIQUE NAME
$: TO GENERATE A UNIQUE FILE NAME IN THE DISK DIRECTORY.  THIS PATCH CHANGES THE
$: METHOD FOR GENERATING A UNIQUE FILE NAME FOR THE ALGOL SOURCE CODE OUTPUT OF
$: THE COMPILER.  FORMERLY, THIS WAS DONE USING THE TIME FUNCTION TO OBTAIN
$: SOME RANDOM DIGITS.  THE METHOD USED IN PATCH/MERGE IS ADOPTED HERE, NAMELY
$: STARTING WITH THE PREFIX (MFID) "PASCOO1", A SEARCH IS PERFORMED TO DETERMINE
$: WHETHER SUCH A FILE NAME IS ALREADY CATALOGUED.  IF SO, 1 IS ADDED AND THE
$: SEARCH REPEATED.  IN ADDITION, THE FILE IS CREATED WITH A SAVE FACTOR
$: (RETENTION PERIOD) OF ZERO DAYS SO THAT A HALT-LOAD WILL REMOVE THE FILE
$: AUTOMATICALLY.
$: SEE PATCH 704.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
PROCEDURE  SEARCHDISKDIRECTORY( F, A );  FILE F;  ARRAY A[0];           20222100
          SEARCH( F, A[*] );   % END OF SEARCHDISKDIRECTORY;            20222200
                                                                        20222300
    CHARPNT := POINTER(CH[0])+7;  CH[0] := " ";                         20322000
CH[0] := "PASCOOO";  CHARPNT := POINTER(CH[0])+5;                       90016000
PASCALGOL.FID := USER := TIME(-1);                                      90017000
DO BEGIN   C:=C+1;  REPLACE CHARPNT BY C FOR 3 DIGITS;                  90018000
    PASCALGOL.MFID := ALGOLNAME := CH[0];                               90019000
    SEARCHDISKDIRECTORY( PASCALGOL, LINES[*] );                         90020000
END UNTIL LINES[0]=-1; % FILE NOT ON DISK                               90021000
$# PATCH 712 FOR PASCAL.XVI.0 CONTAINS 2 CARDS.  MARK PROCEDURE LEVELS IN MARGIN
$: PATCH TO MARK THE START AND END OF PROCEDURES AND FUNCTIONS BY ANNOTATING THE
$: MARGIN WITH THE SYMBOLS "+P" & "-P" FOLLOWED BY THE LEVEL NUMBER.
$:  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$:
    MARGIN("+P",CURLEVEL);   % MARK PROCEDURE LEVEL                     80420100
    MARGIN("-P",CURLEVEL);   % MARK END OF PROCEDURE                     80702100
$# PATCH 713 FOR PASCAL.XVI.0. CONTAINS 14 CARDS.CORRECTS ERROR MESSAGE ETC.
$:   CORRECTS THE DOUBLE "NO ERRORS" MESSAGE AND THE OUTPUT OF HEADINGS
$:   WHEN L1 IS SET AFTER L=.
$:   ALSO CORRECTS THE SCANNING PROBLEM WHEN COMPILER OPTIONS ARE INCORRECT.
$:   DAVID A COOPER, HERIOT-WATT UNIVERSITY ..... AUGUST 1978
```

```
$;
       ERROR102MESS(//"102  **; WARNING ONLY, ILLEGAL COMPILER OPTION.")10188750
       , %                                                              10188751
    IF ERRNUM=100 OR ERRNUM=102                                         20181600
    THEN NUMERRS := NUMERRS - 1; %*ERROR NUMBER 102 IS ONLY AN ILLEGAL  20181610
%                                * DOLLAR OPTION WARNING &              20181620
%                                *ERROR NUMBER 100 ALONE SHOULD NOT     20181650
                                ELSE LISTOPTION := C="+" ELSE           30265000
            END                                                         30280800
            ELSE ERROR(102);                                            30280810
     IF ERR(102) THEN                                                   90090710
         WRITE(LINE,ERROR102MESS);                                      90090720
$                                                                       90110000
$                                                                       90111000
     ("102  ***WARNING ONLY; ILLEGAL DOLLAR OPTION."),                  91106900
$# PATCH 800 FOR PASCAL.XVI.0. CONTAINS 10 CARDS.
$; TO REMOVE CONFLICTS BETWEEN HERIOT-WATT & NATAL EXISTING PATCHES.
$;
       MAXSYMS     = 800#,     %MAX NUMBER OF SYMBOLS IN ONE EXPRESSION.  10052000
       MAXPNTRS    = 25#;      %MAX NUMBER OF UNDECLARED POINTERS(FORWD).  10056000
DEFINE ERR(ERR1) = BOOLEAN(@&ERRP[(ERR1).[6:2]][0:((ERR1).[4:5])+1])#;    10156100
INTEGER EXPRLEVEL, EXPINVARCNT; %                                       40018000
 INTEGER INDEX, CTYPE, NUMFORWARDS,TX, I; %                             80403000
INTEGER PROGNAMELENGTH; %                                               90013900
     IF ERR(100) %                                                      90090600
       "; ALGOL FILE TAPE=PASCRUN/DISK SERIAL; ALGOL FILE CARD=",       90119000
        P(PALGOLNAME),"/",P(PUSER)," SERIAL;", %                        90120000
       " XALGOL STACK = 2048; END."; %                                  90120500
$# PATCH 998 FOR PASCAL.XVI.0 CONTAINS 10 CARDS.  INSERT PAGE THROWS AT DESIRED
$; PATCH TO INSERT PAGE THROWS AT DESIRED POINTS IN THE SOURCE TO PRODUCE A
$; NICELY LAID OUT LISTING.
$;
$                                                       PAGE            19000000
$                                                       PAGE            20290000
$                                                       PAGE            29000000
$                                                       PAGE            39000000
$                                                       PAGE            49000000
$                                                       PAGE            59000000
$                                                       PAGE            69000000
$                                                       PAGE            79000000
$                                                       PAGE            89000000
$                                                       PAGE            90070999
$;  NILS A OTTE,   UNIVERSITY OF NATAL, DURBAN.    AUG - NOV 1977.
$;
$# PATCH 999 FOR PASCAL.XVI.0. CONTAINS 1 CARDS. VERSION NUMBER.
$;
DEFINE EDITION = "4.4"#;%AUGUST    1978...DAVID A COOPER...             10028000


 LABEL  000000000LINE   001790A1CC EX 0/R;COMMON=1;FILE S=PATCHES/PASCAL;FILE LINE=LINE PRINT;END.        0       /R
```

```
LABEL  000000000LINE   001790A1CC EX 0/R;FILE LINE=LINE PRINT;COMMON=1;FILE S=SYMBOL/PASCAL;END*          0      /R
                    FILE: SYMBOL/PASCAL     FRIDAY  03/02/79  03:49 PM

                                                                                                        10001000
                                                                                                        10002000
%%%
```

```
        MAXEXTFILES=20#,        %MAX NUMBER OF EXTERNAL FILES.                  10054000
        MAXFILES   =20#,        %MAX NUMBER OF FILES DECLARED AT ONE TIME.      10055000
        MAXPNTRS   =50#;        %MAX NUMBER OF UNDECLARED POINTERS.             10056000
                                                                                10057000
%*** NAME TABLES ***                                                            10058000
ARRAY NAMETAB1,NAMETAB2,NAMETAB3[0:MAXTABLES,0:MAXNAMES];                       10059000
DEFINE NAMELENGTH  =[41:6]#,                                                    10060000
       TYPE        =[9:10]#,                                                    10061000
       IDCLASS     =[12:3]#,                                                    10062000
             VAR   =0#,                                                         10063000
             CONST=1#,                                                          10064000
             FUNC  =2#,                                                         10065000
             PROC  =3#,                                                         10066000
             TYPES=4#,                                                          10067000
       INFO        =[23:11]#,                                                   10068000
       FORMAL      =[24:1]#,                                                    10069000
       FORWARDDEF  =[25:1]#,                                                    10070000
       EXTERNALFILE=[26:1]#;                                                    10071000
                                                                                10072000
%*** DISPLAY VECTOR ***                                                         10073000
ARRAY DISPLAY[0:MAXLEVEL];                                                      10074000
DEFINE RECTYPE        =[9:10]#,                                                 10075000
       FIRSTWITHSYM   =[19:10]#,                                                10076000
       LASTWITHSYM    =[29:10]#,                                                10077000
       NUMPNTRSINWITH=[35:6]#,                                                  10078000
       BRACKETSINWITH=[36:1]#,                                                  10079000
       NAMETAB        =[46:7]#;                                                 10080000
                                                                                10081000
%*** TYPE TABLES ***                                                            10082000
ARRAY TYPETAB1,TYPETAB2,TYPETAB3[0:MAXTYPES];                                   10083000
DEFINE FORM  =[3:4]#,                                                           10084000
            NUMERIC =0#,                                                        10085000
            SYMBOLIC=1#,                                                        10086000
            SUBTYPE =2#,                                                        10087000
                MAINTYPE=[33:10]#,                                              10088000
            CHAR    =3#,                                                        10089000
            FLOATING=4#,                                                        10090000
            ALFA    =5#,                                                        10091000
            SET     =6#,                                                        10092000
                SETTYPE =[33:10]#,                                              10093000
            POINTERS=7#,                                                        10094000
                POINTTYPE=[33:10]#,                                             10095000
            ARRAYS  =8#,                                                        10096000
                INXTYPE =[33:10]#,                                              10097000
                ARRTYPE =[43:10]#,                                              10098000
            RECORD  =9#,                                                        10099000
                RECTAB  =[33:10]#,                                              10100000
            FILES   =10#,                                                       10101000
                FILETYPE=[33:10]#,                                              10102000
            TEXTFILE=11#,                                                       10103000
       SIZE  =[15:12]#,                                                         10104000
       STRUCT=[23:8]#;                                                          10105000
INTEGER NUMTYPES;                                                               10106000
                                                                                10107000
%*** PARAMETER TABLE ***                                                        10108000
ARRAY PARAMTAB[0:MAXPARAMS];                                                    10109000
DEFINE PARAMNAME =[9:10]#,                                                      10110000
```

```
        PARAMKIND =[13:4]#,                                              10111000
        PARAMLEVEL=[23:10]#,                                             10112000
        PARAMTYPE =[33:10]#,                                             10113000
        PARAMFILE =[34:1]#;                                              10114000
INTEGER NUMPARAMS;                                                       10115000
                                                                         10116000
%*** CONSTANT TABLE ***                                                  10117000
ARRAY CONSTTAB[0:MAXCONSTS];                                             10118000
INTEGER NUMCONSTS;                                                       10119000
                                                                         10120000
%*** LABEL TABLE ***                                                     10121000
ARRAY LABTAB[0:MAXLABS];                                                 10122000
DEFINE LABVAL=[14:15]#,                                                  10123000
       LABDEF=[15:1]#;                                                   10124000
INTEGER NUMLABS,FIRSTLAB;                                                10125000
                                                                         10126000
%*** TABLES FOR I/O AND CHARACTER HANDLING ***                           10127000
ARRAY CH[0:0], TEXT[0:1], STRING[0:11];                                  10128000
  POINTER CHARPNT,TEXTPNT,TEXTPNT0,STRINGPNT;                            10129000
ARRAY CARD[0:9], LINE[0:16], XLINE[0:10], ALGOLCARD[0:9];                10130000
  POINTER CARDPNT,LINEPNT,XLINEPNT,ALGOLPNT;                             10131000
  INTEGER CHARCNT,ALGOLCNT,MARGINCNT;                                    10132000
ARRAY HEADTEXT[0:10], ERRLINE[0:16];                                     10133000
  INTEGER LINECNT,PAGECNT,ERRINX;                                        10134000
                                                                         10135000
%*** XREF FILE AND TABLE ***                                             10136000
FILE XREFFILE DISK SERIAL [20:3000] (2,3,150);                           10137000
ARRAY BLOCKTAB[0:MAXTABLES], XREFLINE[0:16]; %                           10138000
  INTEGER NUMXREF,NUMBLOCKS;  POINTER XREFPNT;                           10139000
%                                                                        10140000
%*** OTHER TABLES ***                                                    10141000
INTEGER ARRAY VARLIST[0:LISTLENGTH];    % TEMPORARY LIST OF VARIABLES.   10142000
  INTEGER VARINDEX,FIRSTVAR;                                             10143000
ARRAY SYMTAB[0:MAXSYMS];                % USED BY "EXPRESSION".          10144000
  INTEGER NUMSYMS;                                                       10145000
ARRAY WITHTAB[0:MAXWITHSYMS];           % USED BY "WITHSTAT".            10146000
  INTEGER NWITHSYMS;                                                     10147000
INTEGER ARRAY SYMBOL[0:64];             % USED BY "INSYMBOL".            10148000
INTEGER ARRAY SYMKIND[0:61];            % USED IN ERROR RECOVERY.        10149000
ARRAY PNTRTAB1,PNTRTAB2,PNTRTAB3[0:MAXPNTRS];% USED FOR FORWARD POINTERS 10150000
  INTEGER NUMPNTRS;                                                      10151000
ARRAY EXTFILETAB[0:MAXEXTFILES];        % EXTERNAL FILES.                10152000
  INTEGER NUMEXTFILES;                                                   10153000
ARRAY FILETAB[0:MAXFILES];              % FILES IN USE.                  10154000
  INTEGER NUMFILES;                                                      10155000
BOOLEAN ARRAY ERR[0:119];               % RECORDS ERROR MESSAGES.        10156000
                                                                         10157000
%*** COMPILE TIME OPTIONS ***                                            10158000
BOOLEAN LISTOPTION,RESWORDOPTION,CHECKOPTION,DUMPOPTION,XREFOPTION;      10159000
INTEGER CARDLENGTH;                                                      10160000
                                                                         10161000
%*** INTRINSIC TYPES ***                                                 10162000
INTEGER INTTYPE,REALTYPE,ALFATYPE,CHARTYPE,BOOLTYPE,NILTYPE,TEXTTYPE,    10163000
  INPUTFILE,OUTPUTFILE,EMPTYSET;                                         10164000
BOOLEAN INPUTDECL,OUTPUTDECL;                                            10165000
                                                                         10166000
%*** TEMPORARY VARIABLES ***                                             10167000
```

```
INTEGER T1,T2,T3,T4,T5;                                                         10168000
                                                                                10169000
%*** OTHER VARIABLES ***                                                        10170000
ALPHA USER;                    % THE USER NUMBER FOUND ON THE  USER CARD.        10171000
                                                                                10172000
INTEGER CURLEVEL,              % CURRENT PROCEDURE LEVEL.                       10173000
        TOPLEVEL,              % TOP LEVEL IN DISPLAY VECTOR.                   10174000
        NUMBEGINS,             % NUMBER OF "BEGIN"S IN THE PROGRAM.             10175000
        NUMCASES,              % NUMBER OF CASE-STATEMENTS IN PROGRAM.          10176000
        NUMREPS,               % NUMBER OF REPEAT-STATEMENTS IN PROGRAM.        10177000
        NUMTEMPS,              % NUMBER OF TEMPORARY VARIABLES IN USE.          10178000
        CURFUNC,               % INDEX OF FUNCTION CURRENTLY COMPILED.          10179000
        CURSY,                 % LAST SYMBOL READ BY SCANNER.                   10180000
        CURTYPE,               % TYPE OF ENTITY LAST COMPILED.                  10181000
        CURMODE,               % CURRENT EXPRESSION MODE.                       10182000
        LASTREC;               % LAST RECORD TABLE DEFINED.                     10183000
                                                                                10184000
LABEL ENDOFINPUT;                                                               10185000
                                                                                10186000
FORMAT NOERRORS ("NO ERRORS DETECTED."),                                        10187000
       ERRORS   (I5," ERRORS DETECTED"/),                                       10188000
       ALIST    ("$ SET LIST SINGLE"),                                          10189000
       NOALIST  ("$ RESET LIST"),                                               10190000
       LASTLINE ("} TERMINATE: END OF PASCAL PROGRAM."),                        10191000
       TERMMESS ("****  END-OF-INPUT.    COMPILATION TERMINATED.");            10192000
MONITOR EXPOVR;=REALOVERFLOW;                                                   10193000
                                                                                10194000
%*** SCANNER SYMBOLS ***                                                        10195000
DEFINE   IDENTIFIER=1#, INTCONST=2#,   REALCONST=3#,  ALFACONST=4#,             10196000
         CHARCONST=5#,  NOTSY=6#,      ASTERISK=7#,   SLASH=8#,                 10197000
         ANDSY=9#,      DIVSY=10#,     MODSY=11#,     PLUS=12#,                 10198000
         MINUS=13#,     ORSY=14#,      LSSSY=15#,     LEQSY=16#,                10199000
         GEQSY=17#,     GTRSY=18#,     NEQSY=19#,     EQLSY=20#,                10200000
         INSY=21#,      LPAR=22#,      RPAR=23#,      LBRACKET=24#,             10201000
         RBRACKET=25#,  DOUBLEDOT=26#, COMMA=27#,     SEMICOLON=28#,            10202000
         DOT=29#,       ARROW=30#,     COLON=31#,     ASSIGNSY=32#,             10203000
         BEGINSY=33#,   ENDSY=34#,     IFSY=35#,      THENSY=36#,               10204000
         ELSESY=37#,    CASESY=38#,    OFSY=39#,      REPEATSY=40#,             10205000
         UNTILSY=41#,   WHILESY=42#,   DOSY=43#,      FORSY=44#,                10206000
         TOSY=45#,      DOWNTOSY=46#,  GOTOSY=47#,    NILSY=48#,                10207000
         TYPESY=49#,    ARRAYSY=50#,   RECORDSY=51#,  FILESY=52#,               10208000
         SETSY=53#,     CONSTSY=54#,   VARSY=55#,     LABELSY=56#,              10209000
         FUNCSY=57#,    PROCSY=58#,    WITHSY=59#,    PROGRAMSY=60#,            10210000
         PACKEDSY=61#;                                                          10211000
                                                                                10212000
DEFINE INITIAL=0#, MIDDLE=1#, TERMINAL=2#;                                      10213000
DEFINE NUMBER=0#, BITPATTERN=1#;                                                10214000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%20001000
%                                                                               %20002000
%                                                                               %20003000
%                                                                               %20004000
%                  PART 2:  COMPILER UTILITY ROUTINES.                          %20005000
%                           -------------------------                           %20006000
%                                                                               %20007000
%                                                                               %20008000
%                                                                               %20009000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%20010000
```

```
                                                                                20011000
                                                                                20012000
PROCEDURE INSYMBOL;     FORWARD;                                                20013000
PROCEDURE WRITEALGOL;   FORWARD;                                                20014000
PROCEDURE NEWXREF(NAME1,NAME2,TABLE,DECL);                                      20015000
VALUE NAME1, NAME2, TABLE, DECL;                                                20016000
REAL NAME1, NAME2;                                                              20017000
INTEGER TABLE;                                                                  20018000
BOOLEAN DECL;                                                                   20019000
FORWARD;                                                                        20020000
                                                                                20021000
DEFINE NDIGITS(N)=                                                              20022000
IF N< 9 THEN 1 ELSE                                                             20023000
IF N<99 THEN 2 ELSE 3 DIGITS#;                                                  20024000
                                                                                20025000
DEFINE HEADING=                                                                 20026000
BEGIN COMMENT *** PRINTS A HEADING ON TOP OF A NEW PAGE. ;                      20027000
    PAGECNT:=PAGECNT+1;                                                         20028000
    REPLACE POINTER(HEADTEXT[*])+85 BY PAGECNT FOR NDIGITS(PAGECNT);            20029000
    WRITE(LINES[PAGE]);                                                         20030000
    WRITE(LINES[DBL],11,HEADTEXT[*]);                                           20031000
    LINECNT:=2;                                                                 20032000
END OF HEADING#;                                                                20033000
                                                                                20034000
                                                                                20035000
DEFINE PRINTLINE=               *** PRINTS A SOURCE CODE LINE.                  20036000
BEGIN                                                                           20037000
    REPLACE LINEPNT=8 BY CARDCNT FOR 5 DIGITS;                                  20038000
    IF LINECNT≥LINESPERPAGE THEN HEADING;                                       20039000
    IF RESWORDOPTION THEN                                                       20040000
    BEGIN                                                                       20041000
      WRITE(LINES[NO],11,XLINE[*]);                                             20042000
      WRITE(LINES[NO],11,XLINE[*]);                                             20043000
    END;                                                                        20044000
    WRITE(LINES,17,LINE[*]);                                                    20045000
    LINECNT:=LINECNT+1;                                                         20046000
END OF PRINTLINE#;                                                              20047000
                                                                                20048000
                                                                                20049000
DEFINE NEWCARD=                 *** READS A NEW SOURCE CODE CARD.               20050000
BEGIN                                                                           20051000
    IF LISTOPTION THEN PRINTLINE;                                               20052000
    IF ERRINX>0 THEN PRINTERRORS;                                               20053000
    READ(CARD,10,ICARD[*]) [ENDOFINPUT];                                        20054000
    CARDPNT:=POINTER(ICARD[*]);                                                 20055000
    REPLACE LINEPNT BY CARDPNT FOR 10 WORDS, " " FOR 6 WORDS;                   20056000
    REPLACE XLINEPNT BY " " FOR 10 WORDS;                                       20057000
    CHARCNT:=CARDLENGTH;                                                        20058000
    MARGINCNT:=85;                                                              20059000
    CARDCNT:=CARDCNT+1;                                                         20060000
END#;                                                                           20061000
                                                                                20062000
                                                                                20063000
DEFINE GEN(T,N,START)=          *** GENERATE A TEXT "T", CONSISTING OF          20064000
BEGIN                           *** "N" LETTERS, STARTING AT "START".           20065000
    IF ALGOLCNT<N THEN WRITEALGOL;                                              20066000
    TEXT[0]:=T;                                                                 20067000
```

```
    REPLACE ALGOLPNT:ALGOLPNT BY TEXTPNT0+START FOR N;                    20068000
    ALGOLCNT:=ALGOLCNT-N;                                                20069000
END#;                                                                    20070000
                                                                         20071000
                                                                         20072000
DEFINE GENID(L,NUM,NDIG)=     %**** GENERATE AN XALGOL IDENTIFIER.       20073000
BEGIN                                                                    20074000
    IF ALGOLCNT≤NDIG THEN WRITEALGOL;                                    20075000
    CH[0]:=L;                                                            20076000
    REPLACE ALGOLPNT:ALGOLPNT BY CHARPNT FOR 1, NUM FOR NDIG DIGITS;     20077000
    ALGOLCNT:=ALGOLCNT-NDIG-1;                                           20078000
END#;                                                                    20079000
                                                                         20080000
                                                                         20081000
INTEGER NABS,NSIZE;                                                      20082000
                                                                         20083000
DEFINE GENINT(N)=                                                        20084000
BEGIN                                                                    20085000
    NABS:=ABS(N); IF N<0 THEN GEN("-",1,7);                              20086000
    NSIZE:=IF NABS≤       9 THEN 1 ELSE                                  20087000
           IF NABS≤     999 THEN 3 ELSE                                  20088000
           IF NABS≤   99999 THEN 5 ELSE                                  20089000
           IF NABS≤99999999 THEN 8 ELSE 12;                              20090000
    IF ALGOLCNT<NSIZE THEN WRITEALGOL;                                   20091000
    IF NSIZE=12 THEN                                                     20092000
    REPLACE ALGOLPNT:ALGOLPNT BY (NABS DIV 1000000) FOR 6 DIGITS,        20093000
                          ENTIER(NABS MOD 1000000) FOR 6 DIGITS ELSE     20094000
    REPLACE ALGOLPNT:ALGOLPNT BY NABS FOR NSIZE DIGITS;                  20095000
    ALGOLCNT:=ALGOLCNT-NSIZE;                                            20096000
END OF GENINT#;                                                          20097000
                                                                         20098000
                                                                         20099000
PROCEDURE GENREAL(X);                                                    20100000
VALUE X; REAL X;                                                         20101000
BEGIN                                                                    20102000
  REAL ABSX;                                                             20103000
  INTEGER POWER,V1,V2;                                                   20104000
                                                                         20105000
  IF X.[46:5]=0 THEN                                                     20106000
  BEGIN                                                                  20107000
    IF ALGOLCNT<9 THEN WRITEALGOL;                                       20108000
    TEXT[0]:=X;                                                          20109000
    REPLACE ALGOLPNT:ALGOLPNT BY """, TEXTPNT FOR 7, """;                20110000
    ALGOLCNT:=ALGOLCNT-9;                                                20111000
  END ELSE                                                               20112000
  BEGIN                                                                  20113000
    IF ALGOLCNT<22 THEN WRITEALGOL;                                      20114000
    IF X<0 THEN GEN("(-",2,6);                                           20115000
    ABSX:=ABS(X);                                                        20116000
    IF ABSX>0 THEN                                                       20117000
    BEGIN                                                                20118000
      WHILE ABSX≥1@7 DO BEGIN ABSX:=ABSX/10; POWER:=POWER+1 END;         20119000
      WHILE ABSX<1@6 DO BEGIN ABSX:=ABSX×10; POWER:=POWER-1 END;         20120000
      V1:=ENTIER(ABSX);                                                  20121000
      V2:=ENTIER((ABSX-V1)×100000000);                                   20122000
      REPLACE ALGOLPNT:ALGOLPNT BY V1 FOR 7 DIGITS, ".",                 20123000
        V2 FOR 8 DIGITS, "@";                                            20124000
```

```
      ALGOLCNT:=ALGOLCNT-15;                                                    20125000
      IF POWER<0 THEN GEN("-",1,7);                                             20126000
      POWER:=ABS(POWER);                                                        20127000
      REPLACE ALGOLPNT:ALGOLPNT BY POWER FOR 2 DIGITS;                          20128000
      ALGOLCNT:=ALGOLCNT-2;                                                     20129000
    END ELSE GEN("0",1,7);                                                      20130000
    IF X<0 THEN GEN(")",1,7);                                                   20131000
  END;                                                                          20132000
END OF GENREAL;                                                                 20133000
                                                                                20134000
                                                                                20135000
INTEGER TYPEINDEX;                                                              20136000
                                                                                20137000
DEFINE NEWTYPE=                                                                 20138000
BEGIN                                                                           20139000
    IF NUMTYPES≥MAXTYPES THEN BEGIN ERROR(45);NUMTYPES:=MAXTYPES-20 END;        20140000
    TYPEINDEX:=NUMTYPES:=NUMTYPES+1;                                            20141000
END #;                                                                          20142000
                                                                                20143000
                                                                                20144000
PROCEDURE WRITEALGOL;          ¢*** WRITES A COMPLETED XALGOL CARD TO           20145000
BEGIN                          ¢*** THE FILE.                                   20146000
    REPLACE POINTER(ALGOLCARD[0]) BY CARDCNT FOR 8 DIGITS;                      20147000
    WRITE(PASCALGOL,10,ALGOLCARD[*]);                                           20148000
    IF DUMPOPTION THEN WRITE(LINES,10,ALGOLCARD[*]);                            20149000
    ALGOLPNT:=POINTER(ALGOLCARD[*]);  ALGOLCNT:=71;                             20150000
    REPLACE ALGOLPNT BY " " FOR 9 WORDS;                                        20151000
END OF WRITEALGOL;                                                              20152000
                                                                                20153000
                                                                                20154000
DEFINE MARGIN(LETTER,NUM)=                                                      20155000
BEGIN COMMENT *** PLACES INFORMATION IN THE MARGIN. ;                           20156000
    IF MARGINCNT≤118 THEN                                                       20157000
    BEGIN TEXT[0]:=LETTER;                                                      20158000
          REPLACE LINEPNT+MARGINCNT BY TEXTPNT+5 FOR 2,                         20159000
                  NUM FOR NDIGITS(NUM);                                         20160000
          MARGINCNT:=MARGINCNT+6;                                               20161000
    END;                                                                        20162000
END OF MARGIN#;                                                                 20163000
                                                                                20164000
                                                                                20165000
PROCEDURE SKIP(SYMBOL);        ¢*** SKIP SYMBOLS TO RECOVER FROM ERROR          20166000
VALUE SYMBOL; INTEGER SYMBOL; ¢*** CONDITION.                                   20167000
BEGIN                                                                           20168000
    WHILE CURSY≠SYMBOL AND SYMKIND[CURSY]=MIDDLE DO                             20169000
    IF CURSY=RECORDSY THEN                                                      20170000
    BEGIN DO BEGIN INSYMBOL;                                                    20171000
                   SKIP(99);                                                    20172000
          END UNTIL CURSY≠SEMICOLON AND CURSY≠CASESY;                           20173000
    END ELSE INSYMBOL;                                                          20174000
END OF SKIP;                                                                    20175000
                                                                                20176000
                                                                                20177000
PROCEDURE ERROR(ERRNUM);                                                        20178000
VALUE ERRNUM;  INTEGER ERRNUM;                                                  20179000
BEGIN COMMENT *** ARRANGE ERROR INDICATOR. ;                                    20180000
    NUMERRS:=NUMERRS+1;                                                         20181000
```

```
    ERR[ERRNUM]:=TRUE;                                                    20182000
    ERRINX:=MAX(ERRINX,CARD(LENGTH-2-CHARCNT));                           20183000
    IF ERRINX≤115 THEN                                                    20184000
    BEGIN REPLACE POINTER(ERRLINE[1])+ERRINX BY "x",                      20185000
          ERRNUM FOR NDIGITS(ERRNUM);                                     20186000
        ERRINX:=ERRINX+(IF ERRNUM≤ 9 THEN 2 ELSE                          20187000
                        IF ERRNUM≤99 THEN 3 ELSE 4);                      20188000
END END OF ERROR;                                                         20189000
                                                                          20190000
                                                                          20191000
PROCEDURE PRINTERRORS;                                                    20192000
BEGIN COMMENT *** PRINT ERROR INDICATORS. ;                               20193000
    IF NOT LISTOPTION THEN PRINTLINE;                                     20194000
    WRITE(LINES,17,ERRLINE[*]);                                           20195000
    LINECNT:=LINECNT+1;                                                   20196000
    REPLACE POINTER(ERRLINE[1]) BY " " FOR 16 WORDS;                      20197000
    ERRINX:=0;                                                            20198000
END OF PRINTERRORS;                                                       20199000
                                                                          20200000
                                                                          20201000
DEFINE HASH(N) = (N).[35:36] MOD MAXNAMES#;                               20202000
                                                                          20203000
INTEGER THISLEVEL,THISTAB,THISINDEX;                                      20204000
ALPHA THISID,TNAME;                                                       20205000
BOOLEAN FOUND;                                                            20206000
                                                                          20207000
DEFINE SEARCHTAB(TAB)=          %*** SEARCH NAME TABLE "TAB" FOR THE      20208000
BEGIN                           %*** IDENTIFIER JUST READ.                20209000
      THISINDEX:=HASH(CURNAME1);                                          20210000
      TNAME:=NAMETAB1[TAB,THISINDEX];                                     20211000
      WHILE (IF TNAME=CURNAME1 THEN NAMETAB2[TAB,THISINDEX]≠CURNAME2      20212000
                               ELSE TNAME≠0) DO                           20213000
      BEGIN                                                               20214000
        THISINDEX:=IF THISINDEX=0 THEN MAXNAMES ELSE THISINDEX-1;         20215000
        TNAME:=NAMETAB1[TAB,THISINDEX];                                   20216000
      END;                                                                20217000
      FOUND:=TNAME≠0;                                                     20218000
    IF XREFOPTION THEN                                                    20219000
    IF FOUND THEN NEWXREF(CURNAME1,CURNAME2,TAB,FALSE); %                 20220000
END OF SEARCHTAB#;                                                        20221000
                                                                          20222000
DEFINE SEARCH=                  %*** SEARCH ALL TABLES CURRENTLY IN USE.  20223000
BEGIN                                                                     20224000
    THISLEVEL:=TOPLEVEL+1;                                                20225000
    DO BEGIN                                                              20226000
      THISLEVEL:=THISLEVEL-1;                                             20227000
      THISTAB:=IF THISLEVEL<CURLEVEL THEN THISLEVEL                       20228000
                                     ELSE DISPLAY[THISLEVEL].NAMETAB;     20229000
      SEARCHTAB(THISTAB);                                                 20230000
    END UNTIL FOUND OR THISLEVEL=0;                                       20231000
    THISID:=NAMETAB3[THISTAB,THISINDEX];                                  20232000
END OF SEARCH #;                                                          20233000
                                                                          20234000
                                                                          20235000
DEFINE NEWNAME(NAME1,NAME2,TAB) =                                         20236000
BEGIN                 %*** ENTER A NEW NAME INTO THE NAME TABLE "TAB".    20237000
      THISINDEX:=HASH(NAME1);                                             20238000
```

```
      TNAME:=NAMETAB1[TAB,THISINDEX];                                          20239000
      WHILE(IF TNAME=NAME1 THEN NAMETAB2[TAB,THISINDEX]≠NAME2                  20240000
                           ELSE TNAME≠0) DO                                    20241000
      BEGIN                                                                    20242000
        THISINDEX:=IF THISINDEX=0 THEN MAXNAMES ELSE THISINDEX-1;              20243000
        TNAME:=NAMETAB1[TAB,THISINDEX];                                        20244000
      END;                                                                     20245000
      IF TNAME≠0 THEN ERROR(2);                                                20246000
      NAMETAB1[TAB,THISINDEX]:=NAME1;                                          20247000
      NAMETAB2[TAB,THISINDEX]:=NAME2;                                          20248000
    IF XREFOPTION THEN NEWXREF(NAME1,NAME2,TAB,TRUE);                          20249000
END OF NEWNAME #;                                                            20250000
                                                                             20251000
                                                                             20300000
PROCEDURE INITIALIZE;                       %*** INITIALIZATION ***          20301000
BEGIN                                       %************************        20302000
    INTEGER T1,T3;                                                           20303000
    ALPHA A;                                                                 20304000
    FILL SYMKIND[*] WITH 28(MIDDLE),TERMINAL,4(MIDDLE),INITIAL,TERMINAL,     20305000
      INITIAL,MIDDLE,TERMINAL,INITIAL,MIDDLE,INITIAL,TERMINAL,INITIAL,       20306000
      MIDDLE,INITIAL,2(MIDDLE),INITIAL,MIDDLE,INITIAL,4(MIDDLE),             20307000
      7(INITIAL),MIDDLE;                                                     20308000
                                                                             20309000
    FILL SYMBOL[*] WITH 10(0),0,ARROW,0,COLON,GTRSY,GEQSY,PLUS,9(0),         20310000
      DOT,LBRACKET,ANDSY,LPAR,ISSSY,ARROW,0,9(0),0,ASTERISK,MINUS,           20311000
      RPAR,SEMICOLON,LEQSY,0,SLASH,8(0),COMMA,0,NEQSY,EQLSY,RBRACKET,        20312000
      0,DOUBLEDOT;                                                           20313000
                                                                             20314000
    LINEPNT:=POINTER(LINE[1]);                                               20315000
    XLINEPNT:=POINTER(XLINE[1]);                                             20316000
    REPLACE LINEPNT-8 BY "     => ", " " FOR 16 WORDS;                       20317000
    REPLACE XLINEPNT-8 BY " " FOR 11 WORDS;                                  20318000
    REPLACE POINTER(ERRLINE[*]) BY "****    ", " " FOR 16 WORDS;             20319000
    ALGOLPNT:=POINTER(ALGOLCARD[*]); ALGOLCNT:=71;                           20320000
    REPLACE ALGOLPNT BY " " FOR 9 WORDS;                                     20321000
    CHARPNT:=POINTER(CH[*])+7;                                               20322000
    TEXTPNT:=POINTER(TEXT[*])+1; TEXTPNT0:=TEXTPNT-1;                        20323000
    REPLACE TEXTPNT BY " " FOR 15;                                           20324000
    STRINGPNT:=POINTER(STRING[*]);                                           20325000
    REPLACE POINTER(HEADTEXT[*]) BY " " FOR 10 WORDS, "PAGE    ";            20326000
    REPLACE POINTER(HEADTEXT[*]) BY "PASCAL(", EDITION, ")/B=5700";          20327000
    TEXT[0]:=TIME(5);                                                        20328000
    REPLACE POINTER(HEADTEXT[*])+45 BY TEXTPNT+3 FOR 2, "/",                 20329000
      TEXTPNT+1 FOR 2, "/", TEXTPNT+5 FOR 2;                                 20330000
    T1:=TIME(1)/3600;                                                        20331000
    REPLACE POINTER(HEADTEXT[*])+57 BY (T1 DIV 60) FOR 2 DIGITS, ":",        20332000
      ENTIER(T1 MOD 60) FOR 2 DIGITS;                                        20333000
    HEADING;                                                                 20334000
                                                                             20335000
    %*** INITIALIZE INTRINSIC TYPES, CONSTANTS ETC. ***                      20336000
                                                                             20337000
    INTTYPE:=T3:=1;                         %*** "INTEGER" ***               20338000
    T1:=NUMERIC; T1.SIZE:=1; T1.STRUCT:=0;                                   20339000
    TYPETAB1[1]:=T1; TYPETAB2[1]:=-MAXINT; TYPETAB3[1]:=MAXINT;              20340000
    NEWNAME("INTEGER","R",0); T3.IDCLASS:=TYPES;                             20341000
    NAMETAB3[0,THISINDEX]:=T3;                                               20342000
    REALTYPE:=T3:=2;                        %*** "REAL" ***                  20343000
```

```
T1.FORM:=FLOATING; TYPETAB1[2]:=T1;                                    20344000
NEWNAME("400REAL",0,0); T3.IDCLASS:=TYPES;                             20345000
NAMETAB3[0,THISINDEX]:=T3;                                             20346000
ALFATYPE:=T3:=3;                        %*** "ALFA" ***                20347000
T1.FORM:=ALFA; TYPETAB1[3]:=T1;                                        20348000
NEWNAME("400ALFA",0,0); T3.IDCLASS:=TYPES;                             20349000
NAMETAB3[0,THISINDEX]:=T3;                                             20350000
BOOLTYPE:=T3:=4;                        %*** "BOOLEAN" ***             20351000
T1.FORM:=SYMBOLIC; TYPETAB1[4]:=T1; TYPETAB3[4]:=1;                    20352000
NEWNAME("7BOOLEA","N",0); T3.IDCLASS:=TYPES;                           20353000
NAMETAB3[0,THISINDEX]:=T3;                                             20354000
CHARTYPE:=T3:=5;                        %*** "CHAR" ***                20355000
T1.FORM:=CHAR; TYPETAB1[5]:=T1; TYPETAB3[5]:=63;                       20356000
NEWNAME("400CHAR",0,0); T3.IDCLASS:=TYPES;                             20357000
NAMETAB3[0,THISINDEX]:=T3;                                             20358000
T3:=BOOLTYPE; T3.IDCLASS:=CONST;    %*** "FALSE" ***                   20359000
NEWNAME("50FALSE",0,0); NAMETAB3[0,THISINDEX]:=T3;                     20360000
T3.INFO:=1;                             %*** "TRUE" ***                20361000
NEWNAME("400TRUE",0,0); NAMETAB3[0,THISINDEX]:=T3;                     20362000
NUMTYPES:=5;                                                           20363000
NILTYPE:=-1;                            %*** TYPE OF "NIL" ***         20364000
EMPTYSET:=-2;                           %*** TYPE OF [] ***            20365000
NEWNAME("6MAXINT",0,0); T3:=INTTYPE;          %*** "MAXINT" ***        20366000
T3.IDCLASS:=CONST; T3.INFO:=1024;                                      20367000
NAMETAB3[0,THISINDEX]:=T3;                                             20368000
NUMCONSTS:=1; CONSTTAB[1]:=MAXINT;                                     20369000
                                                                       20370000
T3:=0; T3.IDCLASS:=PROC;                %*** PROCEDURES ***            20371000
FOR A:="3000GET", "3000NEW", "400PACK", "400PAGE", "3000PUT",          20372000
       "400READ", "6READLN", "50RESET", "6UNPACK", "50WRITE" DO        20373000
BEGIN                                                                  20374000
   NEWNAME(A,0,0); NAMETAB3[0,THISINDEX]:=T3;                          20375000
END;                                                                   20376000
NEWNAME("7DISPOS","F",0); NAMETAB3[0,THISINDEX]:=T3;                   20377000
NEWNAME("7REWRIT","F",0); NAMETAB3[0,THISINDEX]:=T3;                   20378000
NEWNAME("7WRITEL","N",0); NAMETAB3[0,THISINDEX]:=T3;                   20379000
                                                                       20380000
T3.IDCLASS:=FUNC;                       %*** FUNCTIONS ***             20381000
FOR A:="3000ABS", "6ARCTAN", "3000CHR", "3000COS", "3000EOF",          20382000
       "400EOLN", "3000EXP", "20000LN", "3000ODD", "400PRED",          20383000
       "400SUCC", "50ROUND", "3000SIN", "3000SQR", "400SQRT",          20384000
       "50TRUNC", "6CONCAT", "400TIME", "400DATE", "6IOTIME",          20385000
       "400USER", "3000RD"                                             20386000
DO BEGIN                                                               20387000
   NEWNAME(A,0,0); NAMETAB3[0,THISINDEX]:=T3;                          20388000
END;                                                                   20389000
NEWNAME("7EOLPOS","D",0); NAMETAB3[0,THISINDEX]:=T3;                   20390000
NEWNAME("7WEEKDA","Y",0); NAMETAB3[0,THISINDEX]:=T3;                   20391000
                                                                       20392000
TEXTTYPE:=T3:=NUMTYPES:=NUMTYPES+1;  %*** "TEXT" ***                   20393000
T1 := TEXTFILE; T1.STRUCT := 1; TYPETAB1[TEXTTYPE] := T1; %            20394000
T3.IDCLASS := TYPES; %                                                 20395000
NEWNAME("400TEXT",0,0); NAMETAB3[0,THISINDEX]:=T3;                     20396000
T3:=TEXTTYPE; T3.IDCLASS:=VAR;      %*** "INPUT" ***                   20397000
T3.EXTERNALFILE:=1;                                                    20398000
NEWNAME("50INPUT",0,0); INPUTFILE:=THISINDEX;                          20399000
NAMETAB3[0,THISINDEX]:=T3;                                             20400000
```

```
    NEWNAME("6OUTPUT",0,0);                                  %*** "OUTPUT" ***      20401000
    NAMETAB3[0,THISNDEX]:=T3;  OUTPUTFILE:=THISINDEX;                               20402000
END OF INITIALIZE;                                                                  20403000
                                                                                    20404000
                                                                                    20500000
                                                                                    20501000
%***  XREF ROUTINES ***                                                            20502000
%**********************                                                            20503000
                                                                                    20504000
DEFINE XREFCARD=[16:17]#,                                                           20505000
       XREFBLOCK=[26:10]#;                                                          20506000
REAL A0,B0,A1,B1,LASTA0,LASTA1;                                                     20507000
INTEGER NL,LASTBLOCK,A2,AX;                                                         20508000
                                                                                    20509000
PROCEDURE NEWXREF(NAME1,NAME2,TABLE,DECL);                                          20510000
VALUE NAME1,NAME2,TABLE,DECL;                                                       20511000
REAL NAME1,NAME2;                                                                   20512000
INTEGER TABLE;                                                                      20513000
BOOLEAN DECL;                                                                       20514000
BEGIN                                                                               20515000
    NL:=NAME1.NAMELENGTH;                                                           20516000
    IF NL<7 THEN NAME1:=0&NAME1[41:41:6]&NAME1[35:6×NL-1:6×NL]                      20517000
            ELSE NAME2:=0&NAME2[35:6×(NL-6)-1:6×(NL-6)];                            20518000
    AX:=CARDCNT;  AX.XREFBLOCK:=BLOCKTAB[TABLE];                                    20519000
    IF DECL THEN AX:=AX-1000000000000;                                              20520000
    WRITE(XREFFILE,*,NAME1,NAME2,AX);                                               20521000
END OF NEWXREF;                                                                     20522000
                                                                                    20523000
PROCEDURE XREFMAX(A);                                                               20524000
ARRAY A[0];                                                                         20525000
BEGIN                                                                               20526000
    A[0]:="AZZZZZZ";  A[1]:="Z7ZZZZ";  A[2]:=9999999999;                            20527000
END OF XREFMAX;                                                                     20528000
                                                                                    20529000
                                                                                    20530000
BOOLEAN PROCEDURE XREFCOMPARE(A,B);                                                 20531000
ARRAY A,B[0];                                                                       20532000
BEGIN                                                                               20533000
    A0:=A[0];  B0:=B[0];  A1:=A[1];  B1:=B[1];                                      20534000
    XREFCOMPARE:=                                                                   20535000
    IF A0.[35:36]≠B0.[35:36] THEN A0.[35:36]<B0.[35:36] ELSE                        20536000
    IF A1≠B1 THEN A1≤B1 ELSE                                                        20537000
    IF A0≠B0 THEN A0.NAMELENGTH<B0.NAMELENGTH ELSE                                  20538000
    A[2] LEQ B[2];                                                                  20539000
END OF XREFCOMPARE;                                                                 20540000
                                                                                    20541000
                                                                                    20542000
PROCEDURE PRINTXREF(FINIS,A);                                                       20543000
VALUE FINIS;  BOOLEAN FINIS;                                                        20544000
ARRAY A[0];                                                                         20545000
BEGIN                                                                               20546000
    IF FINIS THEN                                                                   20547000
    BEGIN                                                                           20548000
      WRITE(LINES,17,XREFLINE[*]);                                                  20549000
      CLOSE(LINES);                                                                 20550000
      CLOSE(XREFFILE);                                                              20551000
    END                                                                             20552000
```

```
ELSE                                                                    20553000
BEGIN                                                                   20554000
A0:=A[0];   A1:=A[1];   A2:=A[2];                                       20555000
  IF A0=LASTA0 AND A1=LASTA1 AND A2.XREFBLOCK=LASTBLOCK THEN             20556000
  BEGIN                                                                 20557000
    IF NUMXREF=15 THEN                                                  20558000
    BEGIN                                                               20559000
      WRITE(LINES,17,XREFLINE[*]); LINECNT:=LINECNT+1;                  20560000
      IF LINECNT>LINESPERPAGE THEN HEADING;                             20561000
      XREFPNT:=POINTER(XREFLINE[*]); NUMXREF:=0;                        20562000
      REPLACE XREFPNT BY " " FOR 17 WORDS; XREFPNT:=XREFPNT+24;         20563000
    END;                                                                20564000
    REPLACE XREFPNT BY A2.XREFCARD FOR 5 DIGITS;                        20565000
    XREFPNT:=XREFPNT+7; NUMXREF:=NUMXREF+1;                             20566000
  END ELSE                                                              20567000
  IF A2<0 THEN                                                          20568000
  BEGIN                                                                 20569000
    A2:=A2+100000000000;                                                20570000
    WRITE(LINES,17,XREFLINE[*]); LINECNT:=LINECNT+1;                    20571000
    IF LINECNT>LINESPERPAGE THEN HEADING;                               20572000
    XREFPNT:=POINTER(XREFLINE[*]); NUMXREF:=0;                          20573000
    REPLACE XREFPNT BY " " FOR 17 WORDS;                                20574000
    TEXT[0]:=A0.[35:36]; LASTA0:=A0;                                    20575000
    REPLACE XREFPNT BY TEXTPNT+1 FOR A0.NAMELENGTH;                     20576000
    TEXT[0]:=LASTA1:=A1;                                                20577000
    IF A0.NAMELENGTH>6 THEN                                             20578000
    REPLACE XREFPNT+6 BY TEXTPNT+1 FOR A0.NAMELENGTH-6;                 20579000
    REPLACE XREFPNT+17 BY A2.XREFCARD FOR 5 DIGITS;                     20580000
    XREFPNT:=XREFPNT+24; LASTBLOCK:=A2.XREFBLOCK;                       20581000
    END;                                                                20582000
  END;                                                                  20583000
END OF PRINTXREF;                                                       20584000
                                                                        20585000
                                                                        20800000
                                                                        20801000
INTEGER TT1,TT2,F1,F2,LT,RT;                                            20802000
                                                                        20803000
DEFINE CHECKTYPES(LEFTTYPE,RIGHTTYPE)=                                  20804000
BEGIN                                                                   20805000
    IF LEFTTYPE>0 AND RIGHTTYPE>0 THEN                                  20806000
    IF LEFTTYPE≠RIGHTTYPE THEN                                          20807000
    BEGIN                                                               20808000
      LT:=LEFTTYPE;          RT:=RIGHTTYPE;                             20809000
      TT1:=TYPETAB1[LT];     TT2:=TYPETAB1[RT];                         20810000
      F1:=TT1.FORM;          F2:=TT2.FORM;                              20811000
      IF LT≠REALTYPE OR F2≠NUMERIC THEN                                 20812000
      IF(F1≠SET AND LT≠EMPTYSET)OR(F2≠SET AND RT≠EMPTYSET)THEN          20813000
      IF(F1≠POINTERS AND LT≠NILTYPE)OR(F2≠POINTERS AND RT≠NILTYPE)THEN  20814000
      BEGIN                                                             20815000
        IF F1=SET AND F2=SET THEN                                       20816000
        BEGIN                                                           20817000
          LT:=TT1.SETTYPE;       RT:=TT2.SETTYPE;                       20818000
          TT1:=TYPETAB1[LT];     TT2:=TYPETAB1[RT];                     20819000
          F1:=TT1.FORM;          F2:=TT2.FORM;                          20820000
        END;                                                            20821000
        IF F1=POINTERS AND F2=POINTERS THEN                             20822000
        BEGIN                                                           20823000
```

```
          LT:=TT1.POINTTYPE;   RT:=TT2.POINTTYPE;                                  20824000
          TT1:=TYPETAB1[LT];   TT2:=TYPETAB1[RT];                                  20825000
          F1:=TT1.FORM;        F2:=TT2.FORM;                                       20826000
        END;                                                                       20827000
        WHILE F1=SUBTYPE DO                                                        20828000
        BEGIN LT:=TT1.MAINTYPE;  TT1:=TYPETAB1[LT];  F1:=TT1.FORM END;             20829000
        WHILE F2=SUBTYPE DO                                                        20830000
        BEGIN RT:=TT2.MAINTYPE;  TT2:=TYPETAB1[RT];  F2:=TT2.FORM END;             20831000
        IF LT>0 AND RT>0 THEN                                                      20832000
        IF LT#RT THEN                                                              20833000
        IF F1#NUMERIC OR F2#NUMERIC THEN                                           20834000
        IF F1#CHAR OR F2#CHAR THEN ERROR(17);                                      20835000
      END;                                                                         20836000
    END;                                                                           20837000
END OF CHECKTYPES#;                                                                20838000
                                                                                   20839000
                                                                                   20840000
INTEGER FILENAME;                                                                  20841000
BOOLEAN LPARFOUND;                                                                 20842000
                                                                                   20843000
DEFINE FILEPARAM(DEFAULTFILE)=%*** CHECKS THE FIRST PARAMETER TO SEE               20844000
BEGIN                          %*** IF IT IS A FILE.                               20845000
    INSYMBOL;  FILENAME:=CURTYPE:=0;                                               20846000
    LPARFOUND:=CURSY=LPAR;                                                         20847000
    IF LPARFOUND THEN                                                              20848000
    BEGIN                                                                          20849000
      INSYMBOL;                                                                    20850000
      IF CURSY=IDENTIFIER THEN                                                     20851000
      BEGIN                                                                        20852000
        SEARCH;                                                                    20853000
        IF FOUND THEN                                                              20854000
        BEGIN                                                                      20855000
          IF THISID.IDCLASS=VAR THEN                                               20856000
          BEGIN                                                                    20857000
            CURTYPE:=THISID.TYPE;                                                  20858000
            IF TYPETAB1[CURTYPE].FORM≥FILES THEN                                   20859000
            BEGIN                                                                  20860000
              FILENAME:=1000×THISLEVEL+THISINDEX;                                  20861000
              INSYMBOL;                                                            20862000
      END END END END;                                                             20863000
      IF SYMKIND[CURSY]=TERMINAL THEN ERROR(46);                                   20864000
    END;                                                                           20865000
    IF FILENAME=0 THEN FILENAME:=DEFAULTFILE;                                      20866000
    IF (FILENAME=INPUTFILE AND NOT INPUTDECL) OR                                   20867000
       (FILENAME=OUTPUTFILE AND NOT OUTPUTDECL) THEN ERROR(96);                    20868000
END OF FILEPARAM#;                                                                 20869000
                                                                                   20870000
                                                                                   20871000
INTEGER TFORM;                                                                     20872000
BOOLEAN SIGNED,NEGATIVE;                                                           20873000
                                                                                   20874000
DEFINE CONSTANT(CVAL,CTYPE)=                %*** <CONSTANT> ***                    20875000
BEGIN                                       %*******************                   20876000
    IF CURSY=MINUS OR CURSY=PLUS THEN                                              20877000
    BEGIN SIGNED:=TRUE; NEGATIVE:=CURSY=MINUS;                                     20878000
          INSYMBOL;                                                                20879000
    END ELSE SIGNED:=NEGATIVE:=FALSE;                                              20880000
```

```
IF CURSY=INTCONST THEN                                                  20881000
BEGIN CTYPE:=INTTYPE;                                                   20882000
      CVAL:=IF NEGATIVE THEN -CURVAL ELSE CURVAL;                       20883000
END ELSE                                                                20884000
IF CURSY=CHARCONST THEN                                                 20885000
BEGIN IF SIGNED THEN ERROR(29);                                         20886000
      CTYPE:=CHARTYPE;  CVAL:=CURVAL;                                   20887000
END ELSE                                                                20888000
IF CURSY=REALCONST THEN                                                 20889000
BEGIN CTYPE:=REALTYPE;                                                  20890000
      CVAL:=IF NEGATIVE THEN -CURVAL ELSE CURVAL;                       20891000
END ELSE                                                                20892000
IF CURSY=ALFACONST THEN                                                 20893000
BEGIN IF SIGNED THEN ERROR(29);                                         20894000
      IF CURLENGTH>7 THEN ERROR(41);                                    20895000
      CTYPE:=ALFATYPE;  CVAL:=CURVAL;                                   20896000
END ELSE                                                                20897000
IF CURSY=IDENTIFIER THEN                                                20898000
BEGIN                                                                   20899000
  SEARCH;                                                               20900000
  IF FOUND THEN                                                         20901000
  BEGIN                                                                 20902000
   IF THISID.IDCLASS=CONST AND NOT BOOLEAN(THISID.FORMAL) THEN          20903000
    BEGIN                                                               20904000
      IF TYPETAB[THISID.TYPE].FORMSALFA THEN                            20905000
      BEGIN                                                             20906000
        CVAL:=THISID.INFO;                                              20907000
        IF CVAL>1023 THEN CVAL:=CONSTTAB[CVAL-1023];                    20908000
        CTYPE:=THISID.TYPE;                                             20909000
        IF SIGNED THEN                                                  20910000
        BEGIN                                                           20911000
            TFORM:=TYPETAB[THISID.TYPE].FORM;                           20912000
            IF TFORM#NUMERIC AND TFORM#FLOATING THEN ERROR(29) ELSE     20913000
            IF NEGATIVE THEN CVAL:=-CVAL;                               20914000
        END;                                                            20915000
      END ELSE BEGIN ERROR(48); CVAL:=CTYPE:=0 END;                     20916000
    END ELSE BEGIN ERROR(30); CVAL:=CTYPE:=0 END;                       20917000
  END ELSE BEGIN ERROR(1); CVAL:=CTYPE:=0 END;                          20918000
END ELSE BEGIN ERROR(32); CVAL:=CTYPE:=0 END;                           20919000
INSYMBOL;                                                               20920000
END OF CONSTANT#;                                                       20921000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%30001000
%                                                                       %30002000
%                                                                       %30003000
%                                                                       %30004000
%                  PART 3:   THE SCANNER.                               %30005000
%                            ------------                               %30006000
%                                                                       %30007000
%                                                                       %30008000
%                                                                       %30009000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%30010000
                                                                        30011000
%                       INTERNAL      INTERNAL            SYMBOL        30012000
%       SYMBOL           NUMBER         NAME               KIND         30013000
%                                                                       30014000
%       IDENTIFIER         1          IDENTIFIER          MIDDLE        30015000
%       122                2          INTCONST            MIDDLE        30016000
```

```
%       2.5                     3       REALCONST       MIDDLE          30017000
%       "ABCD"                  4       ALFACONST       MIDDLE          30018000
%       "C"                     5       CHARCONST       MIDDLE          30019000
%       NOT                     6       NOTSY           MIDDLE          30020000
%       *                       7       ASTERISK        MIDDLE          30021000
%       /                       8       SLASH           MIDDLE          30022000
%       &       AND             9       ANDSY           MIDDLE          30023000
%       DIV                     10      DIVSY           MIDDLE          30024000
%       MOD                     11      MODSY           MIDDLE          30025000
%       +                       12      PLUS            MIDDLE          30026000
%       -                       13      MINUS           MIDDLE          30027000
%       OR                      14      ORSY            MIDDLE          30028000
%       <       LSS             15      LSSSY           MIDDLE          30029000
%       <=      LEQ     ≤       16      LEQSY           MIDDLE          30030000
%       >=      GEQ     ≥       17      GEQSY           MIDDLE          30031000
%       >       GTR             18      GTRSY           MIDDLE          30032000
%       <>      NEQ     ≠       19      NEQSY           MIDDLE          30033000
%       =       EQL             20      EQLSY           MIDDLE          30034000
%       IN                      21      INSY            MIDDLE          30035000
%       (                       22      LPAR            MIDDLE          30036000
%       )                       23      RPAR            MIDDLE          30037000
%       [                       24      LBRACKET        MIDDLE          30038000
%       ]                       25      RBRACKET        MIDDLE          30039000
%       ..                      26      DOUBLEDOT       MIDDLE          30040000
%       ,                       27      COMMA           MIDDLE          30041000
%       ;                       28      SEMICOLON       TERMINAL        30042000
%       .                       29      DOT             MIDDLE          30043000
%       ↑       @               30      ARROW           MIDDLE          30044000
%       :                       31      COLON           MIDDLE          30045000
%       :=                      32      ASSIGNSY        MIDDLE          30046000
%       BEGIN                   33      BEGINSY         INITIAL         30047000
%       END                     34      ENDSY           TERMINAL        30048000
%       IF                      35      IFSY            INITIAL         30049000
%       THEN                    36      THENSY          MIDDLE          30050000
%       ELSE                    37      ELSESY          TERMINAL        30051000
%       CASE                    38      CASESY          INITIAL         30052000
%       OF                      39      OFSY            MIDDLE          30053000
%       REPEAT                  40      REPEATSY        INITIAL         30054000
%       UNTIL                   41      UNTILSY         TERMINAL        30055000
%       WHILE                   42      WHILESY         INITIAL         30056000
%       DO                      43      DOSY            MIDDLE          30057000
%       FOR                     44      FORSY           INITIAL         30058000
%       TO                      45      TOSY            MIDDLE          30059000
%       DOWNTO                  46      DOWNTOSY        MIDDLE          30060000
%       GOTO                    47      GOTOSY          INITIAL         30061000
%       NIL                     48      NILSY           MIDDLE          30062000
%       TYPE                    49      TYPESY          INITIAL         30063000
%       ARRAY                   50      ARRAYSY         MIDDLE          30064000
%       RECORD                  51      RECORDSY        MIDDLE          30065000
%       FILE                    52      FILESY          MIDDLE          30066000
%       SET                     53      SETSY           MIDDLE          30067000
%       CONST                   54      CONSTSY         INITIAL         30068000
%       VAR                     55      VARSY           INITIAL         30069000
%       LABEL                   56      LABELSY         INITIAL         30070000
%       FUNCTION                57      FUNCSY          INITIAL         30071000
%       PROCEDURE               58      PROCSY          INITIAL         30072000
%       WITH                    59      WITHSY          INITIAL         30073000
```

```
%       PROGRAM         60      PROGRAMSY               INITIAL         30074000
%       PACKED          61      PACKEDSY                MIDDLE          30075000
                                                                        30076000
                                                                        30077000
DEFINE BLANK=48#, EQUAL=61#, QUOTES=63#, DOLLAR=42#,                    30078000
       LETTER(C)=(17≤C AND C≤2=)OR(33≤C AND C≤41)OR(50≤C AND C≤57)#,    30079000
       ALFANUM(C)=(LETTER(C) OR C≤9)#;                                  30080000
                                                                        30081000
REAL CURVAL;                                                            30082000
ALPHA CURNAME1,CURNAME2,C,CX;                                           30083000
INTEGER CURLENGTH,LASTCHARPOS;                                          30084000
BOOLEAN FINIS;                                                          30085000
                                                                        30086000
DEFINE NEXTCHAR=                                                        30087000
BEGIN COMMENT *** READ NEXT CHARACTER. *** ;                            30088000
    IF CHARCNT=0 THEN C:=BLANK ELSE                                     30089000
    BEGIN                                                               30090000
        REPLACE CHARPNT BY CARDPNT:CARDPNT FOR 1;                       30091000
        C:=CH[0]; CHARCNT:=CHARCNT-1;                                   30092000
END END #;                                                              30093000
                                                                        30094000
                                                                        30095000
                                                                        30096000
PROCEDURE INSYMBOL;                                                     30097000
BEGIN COMMENT *** READS THE NEXT SYMBOL. ***;                           30098000
    INTEGER SCALE,EXP;                                                  30099000
    BOOLEAN NEGEXP;                                                     30100000
    LABEL START,OVERFLOW;                                               30101000
                                                                        30102000
START:                                                                  30103000
    IF C=BLANK THEN                                                     30104000
    BEGIN SCAN CARDPNT:CARDPNT FOR CHARCNT:CHARCNT WHILE =" ";          30105000
        IF CHARCNT=0 THEN BEGIN NEWCARD; GO TO START END;               30106000
        NEXTCHAR;                                                       30107000
    END;                                                                30108000
    IF LETTER(C) THEN                                                   30109000
    BEGIN                                                               30110000
        CURLENGTH:=1;  CURNAME1:=C;  CURNAME2:=0;                       30111000
        NEXTCHAR;                                                       30112000
        WHILE ALFANUM(C) AND CURLENGTH<6 DO                             30113000
        BEGIN CURNAME1:=C&CURNAME1[35:29:30];                           30114000
            CURLENGTH:=CURLENGTH+1;  NEXTCHAR;                          30115000
        END;                                                            30116000
        IF CURLENGTH=6 THEN                                             30117000
        BEGIN                                                           30118000
            WHILE ALFANUM(C) AND CURLENGTH<12 DO                        30119000
            BEGIN CURNAME2:=C&CURNAME2[35:29:30];                       30120000
                CURLENGTH:=CURLENGTH+1;  NEXTCHAR;                      30121000
            END;                                                        30122000
            WHILE ALFANUM(C) DO NEXTCHAR;                               30123000
        END;                                                            30124000
        CURNAME1.NAMELENGTH:=CURLENGTH;                                 30125000
        CASE CURLENGTH OF                                               30126000
        BEGIN ;                                                         30127000
          CURSY:=IDENTIFIER;                                            30128000
          CURSY:=IF CURNAME1="  0000IF" THEN IFSY ELSE                  30129000
                 IF CURNAME1="  0000DO" THEN DOSY ELSE                  30130000
```

```
        IF CURNAMF1="?0000TO" THEN TOSY ELSE                          30131000
        IF CURNAMF1="?00000R" THEN ORSY ELSE                          30132000
        IF CURNAMF1="?00000F" THEN OFSY ELSE                          30133000
        IF CURNAMF1="?0000IN" THEN INSY ELSE IDENTIFIER;              30134000
CURSY:=IF CURNAMF1="?000END" THEN ENDSY ELSE                          30135000
        IF CURNAMF1="?000FOR" THEN FORSY ELSE                         30136000
        IF CURNAMF1="?000DIV" THEN DIVSY ELSE                         30137000
        IF CURNAMF1="?000MOD" THEN MODSY ELSE                         30138000
        IF CURNAMF1="?000NIL" THEN NILSY ELSE                         30139000
        IF CURNAMF1="?000AND" THEN ANDSY ELSE                         30140000
        IF CURNAMF1="?000NOT" THEN NOTSY ELSE                         30141000
        IF CURNAMF1="?000VAR" THEN VARSY ELSE                         30142000
        IF CURNAMF1="?000SET" THEN SETSY ELSE                         30143000
        IF CURNAMF1="?000LSS" THEN LSSSY ELSE                         30144000
        IF CURNAMF1="?000LEQ" THEN LEQSY ELSE                         30145000
        IF CURNAMF1="?000GEQ" THEN GEQSY ELSE                         30146000
        IF CURNAMF1="?000GTR" THEN GTRSY ELSE                         30147000
        IF CURNAMF1="?000NEQ" THEN NEQSY ELSE                         30148000
        IF CURNAMF1="?000EQL" THEN EQLSY ELSE IDENTIFIER;             30149000
CURSY:=IF CURNAMF1="^00THEN" THEN THENSY ELSE                         30150000
        IF CURNAMF1="^00ELSE" THEN ELSESY ELSE                        30151000
        IF CURNAMF1="^00WITH" THEN WITHSY ELSE                        30152000
        IF CURNAMF1="^00CASE" THEN CASESY ELSE                        30153000
        IF CURNAMF1="^00GOTO" THEN GOTOSY ELSE                        30154000
        IF CURNAMF1="^00TYPE" THEN TYPESY ELSE                        30155000
        IF CURNAMF1="^00FILE" THEN FILESY ELSE IDENTIFIER;            30156000
CURSY:=IF CURNAMF1="=0BEGIN" THEN BEGINSY ELSE                        30157000
        IF CURNAMF1="=0WHILE" THEN WHILESY ELSE                       30158000
        IF CURNAMF1="=0UNTIL" THEN UNTILSY ELSE                       30159000
        IF CURNAMF1="=0ARRAY" THEN ARRAYSY ELSE                       30160000
        IF CURNAMF1="=0CONST" THEN CONSTSY ELSE                       30161000
        IF CURNAMF1="=0LABEL" THEN LABELSY ELSE IDENTIFIER;           30162000
    CURSY:=IF CURNAMF1="^REPEAT" THEN REPEATSY ELSE                   30163000
        IF CURNAMF1="^DOWNTO" THEN DOWNTOSY ELSE                      30164000
        IF CURNAMF1="^RECORD" THEN RECORDSY ELSE                      30165000
        IF CURNAMF1="^PACKED" THEN PACKEDSY ELSE IDENTIFIER;          30166000
    CURSY:=IF CURNAMF1="?PROGRA" AND CURNAME2="M" THEN PROGRAMSY      30167000
                                  ELSE IDENTIFIER;                    30168000
    CURSY:=IF CURNAMF1="^FUNCTI" AND CURNAME2="ON" THEN FUNCSY        30169000
                                  ELSE IDENTIFIER;                    30170000
    CURSY:=IF CURNAMF1="^PROCED" AND CURNAME2="URE" THEN PROCSY       30171000
                                  ELSE IDENTIFIER;                    30172000
    CURSY:=IDENTIFIER;      % 10 CHARACTERS.                         30173000
    CURSY:=IDENTIFIER;      % 11 CHARACTERS.                         30174000
    CURSY:=IDENTIFIER;      % 12 CHARACTERS.                         30175000
  END OF CASE;                                                        30176000
  IF RESWORDOPTION AND CURSY#IDENTIFIER THEN                          30177000
  BEGIN T1:=CARDLENGTH-CHARCNT-CURLENGTH;                             30178000
      IF CHARCNT=0 THEN CARDPNT:=CARDPNT+1 ELSE T1:=T1-1;             30179000
      REPLACE XLINEPNT:T1 BY CARDPNT-(CURLENGTH+1)                    30180000
                                        FOR CURLENGTH;                30181000
  END;                                                                30182000
END OF LETTER ELSE                                                    30183000
IF C≤9 THEN                                                           30184000
BEGIN                                                                 30185000
  CURVAL:=C;  CURSY:=INTCONST;                                        30186000
  NEXTCHAR;                                                           30187000
```

```
WHILE C≤9 DO BEGIN CURVAL:=10×CURVAL+C; NEXTCHAR END;                    30188000
IF C="." THEN                                                            30189000
BEGIN                                                                    30190000
    NEXTCHAR;                                                            30191000
    IF C≤9 THEN                                                          30192000
    BEGIN CURSY:=REALCONST;                                              30193000
          DO BEGIN CURVAL:=10×CURVAL+C;                                  30194000
                   SCALE:=SCALE-1;  NEXTCHAR;                            30195000
          END UNTIL C>9;                                                 30196000
    END ELSE IF C="." THEN C:=64      % SPECIAL MARK FOR ".."             30197000
                      ELSE ERROR(4);                                     30198000
END;                                                                     30199000
IF C="E" THEN                                                            30200000
BEGIN                                                                    30201000
    CURSY:=REALCONST;  NEXTCHAR;                                         30202000
    IF C="+" OR C="-" THEN BEGIN NEGEXP:=C="-"; NEXTCHAR END;            30203000
    IF C≤9 THEN                                                          30204000
    BEGIN EXP:=C;  NEXTCHAR;                                             30205000
          WHILE C≤9 DO BEGIN EXP:=10×EXP+C; NEXTCHAR END;                30206000
          IF NEGEXP THEN EXP:=-EXP;                                      30207000
    END ELSE ERROR(4);                                                   30208000
    SCALE:=SCALE+EXP;                                                    30209000
END;                                                                     30210000
IF CURSY=REALCONST THEN                                                  30211000
BEGIN                                                                    30212000
  REALOVERFLOW:=OVERFLOW;                                                30213000
  CURVAL:=CURVAL×10↑SCALE;                                               30214000
  REALOVERFLOW:=0;                                                       30215000
END ELSE                                                                 30216000
        IF CURVAL>MAXINT THEN                                            30217000
        BEGIN                                                            30218000
OVERFLOW:   ERROR(14);  CURVAL:=0;  REALOVERFLOW:=0;                     30219000
        END;                                                             30220000
    END OF DIGIT ELSE                                                    30221000
    IF C=QUOTES THEN                                                     30222000
    BEGIN                                                                30223000
        CURSY:=ALFACONST;  CURLENGTH:=0;  NEXTCHAR;                      30224000
        FINIS:=FALSE;                                                    30225000
        DO BEGIN                                                         30226000
            IF C=QUOTES THEN BEGIN NEXTCHAR; FINIS:=C≠QUOTES END ELSE    30227000
            IF CHARCNT=0 THEN BEGIN ERROR(6); FINIS:=TRUE END;           30228000
            IF NOT FINIS THEN                                            30229000
            BEGIN                                                        30230000
                REPLACE STRINGPNT+CURLENGTH BY CHARPNT FOR 1;            30231000
                CURLENGTH:=CURLENGTH+1;                                  30232000
                NEXTCHAR;                                                30233000
        END END UNTIL FINIS;                                             30234000
        IF CURLENGTH=0 THEN ERROR(4) ELSE                                30235000
        IF CURLENGTH=1 THEN                                              30236000
        BEGIN CURSY:=CHARCONST;                                          30237000
              REPLACE CHARPNT BY STRINGPNT FOR 1;  CURVAL:=CH[0];        30238000
        END ELSE                                                         30239000
        IF CURLENGTH≤7 THEN                                              30240000
        BEGIN TEXT[0]:="       ";                                        30241000
              REPLACE TEXTPNT BY STRINGPNT FOR CURLENGTH;                30242000
              CURVAL:=TEXT[0];                                           30243000
        END;                                                             30244000
```

```
END OF STRINGS  ELSE                                                    30245000
BEGIN                                                                   30246000
    CURSY:=SYMBOL[C];  NEXTCHAR;                                        30247000
    IF CURSY=COLON AND C=EQUAL THEN                                     30248000
       BEGIN CURSY:=ASSIGNSY; NEXTCHAR END ELSE                         30249000
    IF CURSY=DOT AND C="." THEN                                         30250000
       BEGIN CURSY:=DOUBLEDOT; NEXTCHAR END ELSE                        30251000
    IF CURSY=LSSSY AND C=EQUAL THEN                                     30252000
       BEGIN CURSY:=LEQSY; NEXTCHAR END ELSE                            30253000
    IF CURSY=LSSSY AND C="S" THEN                                       30254000
       BEGIN CURSY:=NEQSY; NEXTCHAR END ELSE                            30255000
    IF CURSY=GTRSY AND C=EQUAL THEN                                     30256000
       BEGIN CURSY:=GEQSY; NEXTCHAR END ELSE                            30257000
    IF CURSY=LPAR AND C="*" THEN                                        30258000
    BEGIN                          % *** COMMENT ***                    30259000
        NEXTCHAR;                                                       30260000
        IF C=DOLLAR THEN      %  DOLLAR INDICATES COMPILER OPTIONS.     30261000
        DO BEGIN                                                        30262000
          NEXTCHAR; CX:=C;  NEXTCHAR;                                   30263000
          IF CX="L" THEN IF C=1 THEN HEADING                            30264000
                                ELSE LISTOPTION:=C="+"  ELSE            30265000
          IF CX="R" THEN RESWORDOPTION:=C="+"             ELSE          30266000
          IF CX="C" THEN CHECKOPTION:=C="+"               ELSE          30267000
          IF CX="D" THEN DUMPOPTION:=C="+"                ELSE          30268000
          IF CX="X" THEN XREFOPTION:=C="+"                ELSE          30269000
          IF CX="A" THEN                                                30270000
            IF C="+" THEN WRITE(PASCALGOL,ALIST)                        30271000
                     ELSE WRITE(PASCALGOL,NOALIST)   ELSE               30272000
            IF CX="T" THEN                                              30273000
            BEGIN LASTCHARPOS := CHARCNT - CARDLENGTH;                  30274000
                  CARDLENGTH:=10*C;                                     30275000
                  NEXTCHAR; CARDLENGTH:=CARDLENGTH+C;                   30276000
                  IF CARDLENGTH<9 OR CARDLENGTH>80 THEN                 30277000
                  BEGIN ERROR(14); CARDLENGTH:=72 END;                  30278000
                  CHARCNT:=MAX(0,LASTCHARPOS+CARDLENGTH-1);             30279000
            END;                                                        30280000
            NEXTCHAR;                                                   30281000
          END UNTIL C#",";                                              30282000
          FINIS:=FALSE;                                                 30283000
          DO BEGIN                                                      30284000
            IF C#"*" THEN                                               30285000
            SCAN CARDPNT:CARDPNT FOR CHARCNT:CHARCNT UNTIL ="*";        30286000
            IF CHARCNT=0 THEN NEWCARD ELSE                              30287000
            BEGIN NEXTCHAR;                                             30288000
                  WHILE C="*" DO NEXTCHAR;                              30289000
                  FINIS:=C=")";                                         30290000
          END END UNTIL FINIS;                                          30291000
          NEXTCHAR;                                                     30292000
          GO TO START;                                                  30293000
        END OF COMMENT;                                                 30294000
    END;                                                                30295000
END OF INSYMBOL;                                                        30296000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%40001000
%                                                                       %40002000
%                                                                       %40003000
%                                                                       %40004000
%                  PART 4:   EXPRESSION PARSER.                         %40005000
```

```
%                                   --------------------
%
%
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


PROCEDURE EXPRESSION;   FORWARD;
PROCEDURE CONCAT;       FORWARD;

ALPHA TEMPSYM;
REAL SX;
INTEGER EXPRLEVEL,TX;

DEFINE PUTTEXT(T)=
BEGIN
    IF NUMSYMS=MAXSYMS THEN
    BEGIN ERROR(71);
          NUMSYMS:=1;
    END ELSE NUMSYMS:=NUMSYMS+1;
    SYMTAB[NUMSYMS]:=T;
END OF PUTTEXT #;

DEFINE PUTSYM(S)=
BEGIN
    TEMPSYM:=(S)&1[41:5:6];
    PUTTEXT(TEMPSYM);
END OF PUTSYM #;

DEFINE PUTCONST(VAL)=
BEGIN
    PUTTEXT("2000000");
    PUTTEXT(VAL);
END OF PUTCONST #;

DEFINE PUTDUMMY=
BEGIN
    PUTTEXT("3000000");
END OF PUTDUMMY #;

DEFINE PUTID(L,NUM,NUMDIG)=
BEGIN
    TEXT[0]:="      " & L [35:5:6];
    REPLACE TEXTPNT+2 BY NUM FOR NUMDIG DIGITS;
    PUTTEXT(TEXT[0]);
END OF PUTID#;

DEFINE WRITEEXPR=
BEGIN
    FOR T1:=1 STEP 1 UNTIL NUMSYMS DO
    BEGIN
        SX:=SYMTAB[T1];  TX:=SX.[41:6];
        IF TX=0 THEN GEN(SX,7,5) ELSE
        IF TX=3 THEN             ELSE
        IF TX=1 THEN GEN(SX,1,7) ELSE
        BEGIN
            T1:=T1+1;  SX:=SYMTAB[T1];
```

```
            IF SX.[44:6]=0 THEN GENINT(SX) ELSE GENREAL(SX);                    40063000
    END END;                                                                    40064000
    NUMSYMS:=0;                                                                 40065000
END OF WRITEEXPR#;                                                              40066000
                                                                                40067000
                                                                                40068000
DEFINE CHECKEXPR(LLIM,ULIM)=                                                    40069000
BEGIN                                                                           40070000
    PUTTEXT("CHECK(");                                                          40071000
    EXPRESSION;                                                                 40072000
    PUTSYM(","); PUTCONST(LLIM);                                                40073000
    PUTSYM(","); PUTCONST(ULIM);                                                40074000
    PUTSYM(","); PUTCONST(CARDCNT);                                             40075000
    PUTSYM(")");                                                                40076000
END OF CHECKEXPR#;                                                              40077000
                                                                                40078000
                                                                                40079000
BOOLEAN SIMPLEVARIABLE,INSIDEBRACKETS;                                          40080000
INTEGER NUMPOINTERS;                                                            40081000
                                                                                40082000
PROCEDURE VARIABLE;                                                             40083000
BEGIN                                                                           40084000
    INTEGER STARTSYM,LLIM,ULIM;                                                 40085000
    REAL T;                                                                     40086000
    BOOLEAN INBRACKET,INRECORD;                                                 40087000
    LABEL ADDADDR;                                                              40088000
                                                                                40089000
STARTSYM:=NUMSYMS+1;                                                            40090000
IF THISLEVEL>CURLEVEL THEN          % VARIABLE IN FIELD LIST OF                 40091000
BEGIN                               % RECORD USED IN WITH-STATEMENT.            40092000
  T:=DISPLAY[THISLEVEL];                                                        40093000
  T4:=T.FIRSTWITHSYM;  T5:=T.LASTWITHSYM;                                       40094000
  FOR T3:=T4 STEP 1 UNTIL T5 DO PUTTEXT(WITHTAB[T3]);                           40095000
  INRECORD:=TRUE;                                                               40096000
  INBRACKET:=BOOLEAN(T.BRACKETSINWITH);                                         40097000
  NUMPOINTERS:=NUMPOINTERS+T.NUMPNTRSINWITH;                                    40098000
  SIMPLEVARIABLE:=FALSE;                                                        40099000
  CURTYPE:=T.RECTYPE;  T:=TYPETAB1[CURTYPE];                                    40100000
  GO TO ADDADDR;                                                                40101000
END;                                                                            40102000
IF THISLEVEL>1 AND THISLEVEL<CURLEVEL THEN ERROR(5);                            40103000
CURTYPE:=THISID.TYPE;  SIMPLEVARIABLE:=TRUE;                                    40104000
PUTID("V",1000*THISLEVEL+THISINDEX,5);                                          40105000
INSYMBOL;                                                                       40106000
IF CURSY=LBRACKET OR CURSY=DOT OR CURSY=ARROW THEN                              40107000
BEGIN                                                                           40108000
  SIMPLEVARIABLE:=FALSE;                                                        40109000
  DO BEGIN                                                                      40110000
    IF CURSY=LBRACKET THEN                                                      40111000
    BEGIN                                                                       40112000
      IF NOT(INBRACKET OR INRECORD) THEN                                        40113000
      BEGIN PUTSYM("["); INBRACKET:=TRUE END;                                   40114000
      DO BEGIN                                                                  40115000
        T:=TYPETAB1[CURTYPE];                                                   40116000
        LLIM:=TYPETAB2[CURTYPE];  ULIM:=TYPETAB3[CURTYPE];                      40117000
        IF T.FORM#ARRAYS THEN ERROR(12);                                        40118000
        IF INRECORD THEN PUTTEXT("    +(");                                     40119000
```

```
      INSYMBOL;                                                          40120000
      IF CHECKOPTION THEN CHECKEXPR(LLIM,ULIM) ELSE EXPRESSION;          40121000
      CHECKTYPES(T.INXTYPE,CURTYPE);                                     40122000
      CURTYPE:=T.ARRTYPE;                                                40123000
      IF INRECORD THEN                                                   40124000
      BEGIN                                                              40125000
        IF LLIM<0 THEN BEGIN PUTSYM("+"); PUTCONST(-LLIM) END ELSE       40126000
        IF LLIM>0 THEN BEGIN PUTSYM("-"); PUTCONST( LLIM) END;           40127000
        PUTSYM(")");                                                     40128000
        IF TYPETAB1[CURTYPE].SIZE>1 THEN                                 40129000
        BEGIN PUTSYM("x"); PUTCONST(TYPETAB1[CURTYPE].SIZE) END;         40130000
      END ELSE IF TYPETAB1[CURTYPE].STRUCT>0 THEN PUTSYM(",");           40131000
    END UNTIL CURSY#COMMA;                                               40132000
    IF CURSY#RBRACKET THEN                                               40133000
    BEGIN ERROR(59);  SKIP(RBRACKET);                                    40134000
          IF CURSY=RBRACKET THEN INSYMBOL;                               40135000
    END ELSE INSYMBOL;                                                   40136000
  END OF BRACKETS ELSE                                                   40137000
  IF CURSY=DOT THEN                                                      40138000
  BEGIN                                                                  40139000
    IF NOT(INBRACKET OR INRECORD) THEN                                   40140000
    BEGIN PUTSYM("["); INBRACKET:=TRUE END;                              40141000
    T:=TYPETAB1[CURTYPE];                                                40142000
    IF T.FORM#RECORD THEN ERROR(12);                                     40143000
    INSYMBOL;                                                            40144000
    IF CURSY=IDENTIFIER THEN                                             40145000
    BEGIN                                                                40146000
      SEARCHTAB(T.RECTAB);                                               40147000
      IF FOUND THEN                                                      40148000
        BEGIN                                                            40149000
          THISID:=NAMETAB3[T.RECTAB,THISINDEX];                          40150000
ADDADDR:  PUTSYM("+");                                                   40151000
          PUTCONST(THISID.INFO);  CURTYPE:=THISID.TYPE;                  40152000
        END ELSE BEGIN ERROR(1); CURTYPE:=0 END;                         40153000
      END ELSE BEGIN ERROR(9); CURTYPE:=0 END;                           40154000
      INRECORD:=TRUE;                                                    40155000
      INSYMBOL;                                                          40156000
    END OF DOT ELSE                                                      40157000
    BEGIN  % CURSY=ARROW                                                 40158000
      T:=TYPETAB1[CURTYPE];                                              40159000
      IF T.FORM=FILES THEN                                               40160000
      BEGIN                                                              40161000
        CURTYPE:=T.FILETYPE;                                             40162000
        IF TYPETAB1[CURTYPE].STRUCT=0 THEN PUTTEXT("    [0]");           40163000
      END ELSE                                                           40164000
      IF T.FORM=TEXTFILE THEN                                            40165000
      BEGIN                                                              40166000
        SYMTAB[NUMSYMS]:=SYMTAB[NUMSYMS] & "I" [35:5:6];                 40167000
        PUTSYM("."); PUTTEXT("LASTCH");                                  40168000
        CURTYPE:=CHARTYPE;                                               40169000
      END ELSE                                                           40170000
      IF T.FORM=POINTERS THEN                                            40171000
      BEGIN                                                              40172000
        IF INBRACKET THEN PUTSYM("]");                                   40173000
        INBRACKET:=FALSE;                                                40174000
        IF NUMSYMS+2<=MAXSYMS THEN                                       40175000
        BEGIN                                                            40176000
```

```
                FOR T1:=NUMSYMS STEP -1 UNTIL STARTSYM DO                          40177000
                SYMTAB[T1+2]:=SYMTAB[T1];                                          40178000
                SYMTAB[STARTSYM]:=" MEM[";                                         40179000
                SYMTAB[STARTSYM+1]:=" (T:=";                                       40180000
                NUMSYMS:=NUMSYMS+2; NUMPOINTERS:=NUMPOINTERS+1;                    40181000
                INRECORD:=TRUE;                                                    40182000
              END ELSE ERROR(63);                                                  40183000
              CURTYPE:=T.POINTTYPE;                                                40184000
            END ELSE BEGIN ERROR(12); CURTYPE:=0 END;                              40185000
            INSYMBOL;                                                              40186000
          END OF ARROW;                                                            40187000
        END UNTIL CURSY#LBRACKET AND CURSY#DOT AND CURSY#ARROW;                    40188000
        IF TYPETAB[CURTYPE].STRUCT=0 THEN                                          40189000
        BEGIN                                                                      40190000
          IF INBRACKET THEN PUTSYM("]");                                           40191000
          WHILE NUMPOINTERS>0 DO                                                   40192000
          BEGIN PUTTEXT("=1)DIV");  PUTTEXT(" 1022,");                             40193000
                PUTTEXT(" T MOD");  PUTTEXT(" 1022]");                             40194000
                NUMPOINTERS:=NUMPOINTERS-1;                                        40195000
          END;                                                                     40196000
        END;                                                                       40197000
      END;                                                                         40198000
      INSIDEBRACKETS:=INBRACKET;                                                   40199000
      CURMODE:=NUMBER;                                                             40200000
END OF VARIABLE;                                                                   40201000
                                                                                   40202000
                                                                                   40203000
PROCEDURE PASSPARAMS;                                                              40204000
BEGIN                                                                              40205000
    INTEGER NPARS,PARAM,PTYPE,P,FIRSTSYM;                                          40206000
    BOOLEAN FORMALPROC,CHECK;                                                      40207000
    LABEL EXIT;                                                                    40208000
                                                                                   40209000
    PUTID("V",1000*THISLEVEL+TOTSINDEX,5);                                         40210000
    P:=THISID.INFO;                                                                40211000
    FORMALPROC:=BOOLEAN(THISID.FORMAL);                                            40212000
    NPARS:=PARAMTAB[P]; P:=P+1;                                                    40213000
    IF FORMALPROC THEN NPARS:=0999;                                                40214000
    INSYMBOL;                                                                      40215000
    IF CURSY=LPAR THEN                                                             40216000
    BEGIN                                                                          40217000
      PUTSYM("(");                                                                 40218000
      DO BEGIN                                                                     40219000
        INSYMBOL;                                                                  40220000
        IF NPARS=0 THEN BEGIN ERROR(3); SKIP(RPAR); GO TO EXIT END;                40221000
        PARAM:=PARAMTAB[P]; P:=P+1;                                                40222000
        PTYPE:=PARAM.PARAMTYPE;                                                    40223000
        IF PARAM.PARAMKIND=CONST THEN                                              40224000
        BEGIN                                                                      40225000
          CHECK:=CHECKOPTION AND TYPETAB1[PTYPE].FORM LEQ CHAR;                    40226000
          IF CHECK THEN PUTTEXT("CHECK(");                                         40227000
          PUTDUMMY; FIRSTSYM:=NUMSYMS;                                             40228000
          EXPRLEVEL:=EXPRLEVEL+1;                                                  40229000
          EXPRESSION; EXPRLEVEL:=EXPRLEVEL-1;                                      40230000
          IF CURMODE=BITPATTERN THEN                                               40231000
          BEGIN SYMTAB[FIRSTSYM]:=" REAL("; PUTSYM(")"); END;                      40232000
          IF CHECK THEN                                                            40233000
```

```
          BEGIN
            PUTSYM(","); PUTCONST(TYPETAB2[PTYPE]);
            PUTSYM(","); PUTCONST(TYPETAB3[PTYPE]);
            PUTSYM(","); PUTCONST(CARDCNT); PUTSYM(")");
          END;
        END ELSE
        IF PARAM.PARAMKIND=VAR THEN
        BEGIN
          IF CURSY=IDENTIFIER THEN
          BEGIN
            SEARCH;
            IF FOUND THEN
            BEGIN
              IF THISID.IDCLASS=VAR OR
                 THISID.IDCLASS=CONST AND BOOLEAN(THISID.FORMAL) THEN
              BEGIN
                IF PARAM.PARAMFILE=1 THEN
                BEGIN
                  CURTYPE:=THISID.TYPE;
                  PUTID("V",1000×THISLEVEL+THISINDEX,5);  PUTSYM(",");
                  PUTID("F",1000×THISLEVEL+THISINDEX,5);  PUTSYM(",");
                  PUTID("I",1000×THISLEVEL+THISINDEX,5);
                  INSYMBOL;
                END ELSE
                BEGIN
                  VARIABLE;
                  IF TYPETAB1[CURTYPE].STRUCT>0 THEN
                  IF NOT SIMPLEVARIABLE THEN ERROR(92);
                END;
              END ELSE BEGIN ERROR(8); CURTYPE:=0 END;
            END ELSE BEGIN ERROR(1); CURTYPE:=0 END;
          END ELSE BEGIN ERROR(9); CURTYPE:=0 END;
        END ELSE
        BEGIN
          IF CURSY=IDENTIFIER THEN
          BEGIN
            SEARCH;
            IF FOUND THEN
            BEGIN
              IF THISID.IDCLASS≠PARAM.PARAMKIND THEN ERROR(91);
              PUTID("V",1000×THISLEVEL+THISINDEX,5);
              CURTYPE:=IF THISID.IDCLASS=FUNC THEN THISID.TYPE ELSE 0;
              INSYMBOL;
            END ELSE BEGIN ERROR(1); CURTYPE:=0 END;
          END ELSE BEGIN ERROR(9); CURTYPE:=0 END;
        END;
        CHECKTYPES(PTYPE,CURTYPE);
        NPARS:=NPARS-1;
        IF CURSY=COMMA THEN PUTSYM(",");
      END UNTIL CURSY≠COMMA;
      IF CURSY≠RPAR THEN BEGIN ERROR(89); SKIP(RPAR) END;
EXIT: PUTSYM(")");
      IF CURSY=RPAR THEN INSYMBOL;
    END;
    IF NPARS>0 AND NOT FORMALPROC THEN ERROR(3);
    CURMODE:=NUMBER;
END OF PASSPARAMS;
```

```
                                                                                40291000
                                                                                40292000
PROCEDURE FACTOR;                                %*** FACTOR ***                40293000
BEGIN                                            %***************               40294000
    INTEGER STARTSYM,STYPE,T;                                                   40295000
    BOOLEAN FIRST;                                                              40296000
    REAL VAL;                                                                   40297000
                                                                                40298000
    DEFINE PARAMETER=          %**** CHECK THAT THE FUNCTION HAS 1 PARAM.       40299000
    BEGIN                                                                       40300000
        INSYMBOL;                                                               40301000
        IF CURSY=LPAR THEN                                                      40302000
        BEGIN                                                                   40303000
          PUTSYM("(");  INSYMBOL;  EXPRESSION;                                  40304000
          IF TYPETAB[CURTYPE].FORM=NUMERIC THEN CURTYPE:=INTTYPE;               40305000
          IF CURSY#RPAR THEN BEGIN ERROR(3); SKIP(RPAR) END;                    40306000
          PUTSYM(")");  IF CURSY=RPAR THEN INSYMBOL;                            40307000
        END ELSE ERROR(3);                                                      40308000
    END OF PARAMETER#;                                                          40309000
                                                                                40310000
    CURMODE:=NUMBER;                                                            40311000
    IF CURSY=IDENTIFIER THEN                                                    40312000
    BEGIN                                                                       40313000
      SEARCH;                                                                   40314000
      IF FOUND THEN                                                             40315000
      BEGIN                                                                     40316000
        IF THISID.IDCLASS=VAR OR                                                40317000
           THISID.IDCLASS=CONST AND BOOLEAN(THISID.FORMAL)                      40318000
        THEN VARIABLE ELSE                                                      40319000
        IF THISID.IDCLASS=CONST THEN                                            40320000
        BEGIN                                                                   40321000
          IF THISID.INFO≤1023 THEN PUTCONST(THISID.INFO)                        40322000
                              ELSE PUTCONST(CONSTTAB[THISID.INFO-1023]);        40323000
          CURTYPE:=THISID.TYPE;  CURMODE:=NUMBER;                               40324000
          INSYMBOL;                                                             40325000
        END ELSE                                                                40326000
        IF THISID.IDCLASS=FUNC THEN                                             40327000
        BEGIN                                                                   40328000
          IF THISTAB=0 THEN              %*** INTRINSIC FUNCTION ***            40329000
          BEGIN                                                                 40330000
            INTEGER DUMMY;                                                      40350000
            IF CURNAME:="3000ABS" THEN                % "ABS"                    40351000
            BEGIN                                                               40352000
              PUTTEXT("   ABS");  PARAMETER;                                    40353000
              IF CURTYPE#REALTYPE AND CURTYPE#INTTYPE THEN ERROR(67);           40354000
            END ELSE                                                            40355000
            IF CURNAME:="3000CHR" THEN                % "CHR"                    40356000
            BEGIN                                                               40357000
              INSYMBOL;                                                         40358000
              IF CURSY=LPAR THEN                                                40359000
              BEGIN INSYMBOL; CHECKEXPR(0,63);                                  40360000
                    IF TYPETAB[CURTYPE].FORM#NUMERIC THEN ERROR(67);            40361000
                    IF CURSY#RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;         40362000
                    IF CURSY=RPAR THEN INSYMBOL;                                40363000
              END ELSE ERROR(58);                                               40364000
              CURTYPE:=CHARTYPE;                                                40365000
            END ELSE                                                            40366000
```

```
IF CURNAME1="3000EOF" OR                % "EOF"/"EOLN"          40367000
   CURNAME1="4000EOLN" THEN                                     40368000
BEGIN                                                           40369000
  FIRST:=CURNAME1="3000EOF";                                    40370000
  FILEPARAM(INPUTFILE);                                         40371000
  PUTID("I",FILENAME,5);                                        40372000
  PUTTEXT(IF FIRST THEN "  .EOF" ELSE " .EOLN");                40373000
  IF LPARFOUND THEN                                             40374000
  BEGIN                                                         40375000
    IF CURSY#RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;         40376000
    IF CURSY=RPAR THEN INSYMBOL;                                40377000
  END;                                                          40378000
  CURTYPE:=BOOLTYPE;                                            40379000
END ELSE                                                        40380000
IF CURNAME1="3000ODD" THEN              % "ODD"                 40381000
BEGIN                                                           40382000
  PUTTEXT("   ODD"); PARAMETER;                                 40383000
  IF CURTYPE#INTTYPE THEN ERROR(67);                            40384000
  CURTYPE:=BOOLTYPE; CURMODE:=BITPATTERN;                       40385000
END ELSE                                                        40386000
IF CURNAME1="3000ORD" THEN              % "ORD"                 40387000
BEGIN                                                           40388000
  PUTSYM("("); INSYMBOL;                                        40389000
  IF CURSY=LPAR THEN                                            40390000
  BEGIN                                                         40391000
    INSYMBOL; EXPRESSION;                                       40392000
    IF TYPETAB1[CURTYPE].FORM>CHAR THEN ERROR(67);              40393000
    IF CURSY#RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;         40394000
    INSYMBOL;                                                   40395000
  END ELSE ERROR(58); PUTSYM(")");                              40396000
  CURTYPE:=INTTYPE;                                             40397000
END ELSE                                                        40398000
IF CURNAME1="4000PRED" OR               % "PRED"/"SUCC"         40399000
   CURNAME1="4000SUCC" THEN                                     40400000
BEGIN                                                           40401000
  FIRST:=CURNAME1="4000PRED";                                   40402000
  PUTTEXT("CHECK("); INSYMBOL;                                  40403000
  IF CURSY=LPAR THEN                                            40404000
  BEGIN                                                         40405000
    INSYMBOL; EXPRESSION;                                       40406000
    PUTSYM(IF FIRST THEN "-" ELSE "+"); PUTSYM("1");            40407000
    IF TYPETAB1[CURTYPE].FORM>CHAR THEN ERROR(67);              40408000
    PUTSYM(","); PUTCONST(TYPETAB2[CURTYPE]);                   40409000
    PUTSYM(","); PUTCONST(TYPETAB3[CURTYPE]);                   40410000
    PUTSYM(","); PUTCONST(CARDCNT);                             40411000
    PUTSYM(")");                                                40412000
    IF CURSY#RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;         40413000
    IF CURSY=RPAR THEN INSYMBOL;                                40414000
  END ELSE BEGIN ERROR(58); CURTYPE:=0 END;                     40415000
END ELSE                                                        40416000
IF CURNAME1="5000ROUND" THEN            % "ROUND"               40417000
BEGIN                                                           40418000
  PUTTEXT(" ROUND"); PARAMETER;                                 40419000
  IF CURTYPE#REALTYPE THEN ERROR(67);                           40420000
  NUMSYMS:=NUMSYMS-1; PUTSYM(",");                              40421000
  PUTCONST(CARDCNT); PUTSYM(")");                               40422000
  CURTYPE:=INTTYPE;                                             40423000
```

```
END ELSE                                                                40424000
IF CURNAME1="3000SQR" THEN              % "SQR"                         40425000
BEGIN                                                                   40426000
  PUTTEXT("   SQR");  PARAMETER;                                        40427000
  NUMSYMS:=NUMSYMS-1;  PUTSYM(",");                                     40428000
  PUTCONST(CARDCNT);  PUTSYM(")");                                      40429000
  IF CURTYPE#REALTYPE AND CURTYPE#INTTYPE THEN ERROR(67);               40430000
END ELSE                                                                40431000
IF CURNAME1="50TRUNC" THEN              % "TRUNC"                       40432000
BEGIN                                                                   40433000
  PUTTEXT(" TRUNC");  PARAMETER;                                        40434000
  NUMSYMS:=NUMSYMS-1;  PUTSYM(",");                                     40435000
  PUTCONST(CARDCNT);  PUTSYM(")");                                      40436000
  IF CURTYPE#REALTYPE THEN ERROR(67);                                   40437000
  CURTYPE:=INTTYPE;                                                     40438000
END ELSE                                                                40439000
IF CURNAME1="6CONCAT" THEN              % "CONCAT"                      40440000
CONCAT ELSE                                                             40441000
IF CURNAME1="400TIME" THEN              % "TIME"                        40442000
BEGIN                                                                   40443000
  PUTTEXT("(TIME(");  PUTTEXT("1)/60)");                                40444000
  CURTYPE:=REALTYPE;  INSYMBOL;                                         40445000
END ELSE                                                                40446000
IF CURNAME1="400DATE" THEN              % "DATE"                        40447000
BEGIN                                                                   40448000
  PUTTEXT("CURDAT");                                                    40449000
  CURTYPE:=ALFATYPE;  INSYMBOL;                                         40450000
END ELSE                                                                40451000
IF CURNAME1="7ELAPSED" AND CURNAME2="D" THEN    % "ELAPSED"             40452000
BEGIN                                                                   40453000
  PUTTEXT("(TIME(");  PUTTEXT("2)/60)");                                40454000
  CURTYPE:=REALTYPE;  INSYMBOL;                                         40455000
END ELSE                                                                40456000
IF CURNAME1="6IOTIME" THEN              % "IOTIME"                      40457000
BEGIN                                                                   40458000
  PUTTEXT("(TIME(");  PUTTEXT("3)/60)");                                40459000
  CURTYPE:=REALTYPE;  INSYMBOL;                                         40460000
END ELSE                                                                40461000
IF CURNAME1="7WEEKDAY" AND CURNAME2="Y" THEN    % "WEEKDAY"             40462000
BEGIN                                                                   40463000
  PUTTEXT("WEEKDA");                                                    40464000
  CURTYPE:=ALFATYPE;  INSYMBOL;                                         40465000
END ELSE IF CURNAME1="400USER" THEN    % "USER"                         40466000
BEGIN                                                                   40467000
  PUTTEXT("  TIME");  PUTTEXT("  (-1)");                                40468000
  CURTYPE:=ALFATYPE;  INSYMBOL;                                         40469000
END ELSE                                        % "SIN","COS" ETC.      40470000
BEGIN                                                                   40471000
  PUTTEXT(IF CURNAME1="3000SIN" THEN "   SIN" ELSE                      40472000
          IF CURNAME1="3000COS" THEN "   COS" ELSE                      40473000
          IF CURNAME1="6ARCTAN" THEN "ARCTAN" ELSE                      40474000
          IF CURNAME1="400SQRT" THEN "  SQRT" ELSE                      40475000
          IF CURNAME1="3000EXP" THEN "   EXP" ELSE                      40476000
            "    LN");                                                  40477000
  PARAMETER;                                                            40478000
  IF CURTYPE#REALTYPE AND CURTYPE#INTTYPE THEN ERROR(67);               40479000
  CURTYPE:=REALTYPE;                                                    40480000
```

```
        END;                                                                    40481000
      END OF INTRINSIC FUNCTIONS ELSE                                           40482000
      BEGIN                                                                     40483000
        T:=THISID.TYPE;                                                         40484000
        PASSPARAMS;                                                             40485000
        CURTYPE:=T;                                                             40486000
      END;                                                                      40487000
    END OF FUNCTIONS ELSE                                                       40488000
    IF THISID.IDCLASS=PROC THEN                                                 40489000
    BEGIN                                                                       40490000
      ERROR(68); PASSPARAMS;                                                    40491000
      CURTYPE:=0;                                                               40492000
    END ELSE BEGIN ERROR(66); CURTYPE:=0; INSYMBOL END;                         40493000
  END ELSE BEGIN ERROR(1); CURTYPE:=0; INSYMBOL END;                            40494000
END OF IDENTIFIER ELSE                                                          40495000
IF CURSY≤CHARCONST THEN                                                         40496000
BEGIN                                                                           40497000
  CONSTANT(VAL,CURTYPE); PUTCONST(VAL);                                         40498000
END ELSE                                                                        40499000
IF CURSY=NOTSY THEN                                                             40500000
BEGIN                                                                           40501000
  PUTTEXT("  NOT "); PUTDUMMY; STARTSYM:=NUMSYMS;                               40502000
  INSYMBOL; FACTOR;                                                             40503000
  IF CURTYPE>0 THEN                                                             40504000
  IF CURTYPE≠BOOLTYPE THEN BEGIN ERROR(17); CURTYPE:=0 END;                     40505000
  IF CURMODE=NUMBER THEN                                                        40506000
  BEGIN SYMTAB[STARTSYM]:="    B("; PUTSYM(")");                                40507000
        CURMODE:=BITPATTERN;                                                    40508000
  END;                                                                          40509000
END ELSE                                                                        40510000
IF CURSY=NILSY THEN                                                             40511000
BEGIN                                                                           40512000
  PUTCONST(0); CURTYPE:=NILTYPE;                                                40513000
  INSYMBOL;                                                                     40514000
END ELSE                                                                        40515000
IF CURSY=LPAR THEN                                                              40516000
BEGIN                                                                           40517000
  PUTSYM("(");                                                                  40518000
  INSYMBOL; EXPRESSION;                                                         40519000
  IF CURSY≠RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;                           40520000
  PUTSYM(")");                                                                  40521000
  INSYMBOL;                                                                     40522000
END ELSE                                                                        40523000
IF CURSY=LBRACKET THEN                  %*** SET CONSTANT ***                   40524000
BEGIN                                                                           40525000
  INSYMBOL;                                                                     40526000
  IF CURSY=RBRACKET THEN                                                        40527000
  BEGIN                                                                         40528000
    PUTCONST(0); CURTYPE:=EMPTYSET; CURMODE:=NUMBER;                            40529000
    INSYMBOL;                                                                   40530000
  END ELSE                                                                      40531000
  BEGIN                                                                         40532000
    FIRST:=TRUE;                                                                40533000
    DO BEGIN                                                                    40534000
      IF FIRST THEN FIRST:=FALSE ELSE INSYMBOL;                                 40535000
      PUTTEXT("  BIT("); STARTSYM:=NUMSYMS;                                    40536000
      EXPRESSION;                                                               40537000
```

```
            IF STYPE=0 THEN                                                    40538000
            BEGIN STYPE:=CURTYPE;                                              40539000
                  IF TYPETAB[CURTYPE].FORM>CHAR THEN ERROR(72);                40540000
            END ELSE CHECKTYPES(STYPE,CURTYPE);                                40541000
            IF CURSY=DOUBLEDOT THEN                                            40542000
            BEGIN                                                              40543000
              PUTSYM(",");  SYMTAB[STARTSYM]:=" BITS(";                        40544000
              INSYMBOL;  EXPRESSION;                                           40545000
              IF STYPE=0 THEN                                                  40546000
              BEGIN STYPE:=CURTYPE;                                            40547000
                    IF TYPETAB[CURTYPE].FORM>CHAR THEN ERROR(72);              40548000
              END ELSE CHECKTYPES(STYPE,CURTYPE);                              40549000
            END;                                                               40550000
            PUTSYM(",");  PUTCONST(CARDCNT);  PUTSYM(")");                    40551000
            IF CURSY=COMMA THEN PUTTEXT("    OR");                             40552000
          END UNTIL CURSY≠COMMA;                                               40553000
          IF CURSY≠RBRACKET THEN                                               40554000
          BEGIN ERROR(59);  SKIP(RBRACKET);                                    40555000
                IF CURSY=RBRACKET THEN INSYMBOL;                               40556000
          END ELSE INSYMBOL;                                                   40557000
          NEWTYPE;  T1:=SET;  T1.SIZE:=1;  T1.STRUCT:=0;                       40558000
          T1.SETTYPE:=STYPE;  TYPETAB[TYPEINDEX]:=T1;                          40559000
          CURTYPE:=TYPEINDEX;                                                  40560000
          CURMODE:=BITPATTERN;                                                 40561000
        END;                                                                   40562000
      END OF SET CONSTANT ELSE BEGIN ERROR(99); INSYMBOL END;                  40563000
END OF FACTOR;                                                                 40564000
                                                                               40565000
                                                                               40566000
PROCEDURE TERM;                              %*** TERM ***                     40567000
BEGIN                                        %*************                    40568000
    INTEGER STARTSYM,MODE,TYPE1,MULOPTR,F;                                     40569000
    PUTDUMMY;  STARTSYM:=NUMSYMS;                                              40570000
    FACTOR;                                                                    40571000
    MODE:=CURMODE;                                                             40572000
    WHILE CURSY>ASTERISK AND CURSY≤MODSY DO  % "*","/","DIV","MOD","AND"       40573000
    BEGIN                                                                      40574000
      TYPE1:=CURTYPE;  MULOPTR:=CURSY;                                         40575000
      F:=TYPETAB[TYPE1].FORM;                                                  40576000
      IF F=NUMERIC OR F=FLOATING THEN                                          40577000
      BEGIN                                                                    40578000
        MODE:=NUMBER;                                                          40579000
        IF CURSY=ASTERISK THEN PUTSYM("×") ELSE                                40580000
        IF CURSY=SLASH    THEN PUTSYM("/") ELSE                                40581000
        IF CURSY=ANDSY    THEN ERROR(64)   ELSE                                40582000
        BEGIN                                                                  40583000
          IF F=FLOATING THEN ERROR(64);                                        40584000
          IF CURSY=DIVSY THEN PUTTEXT("   DIV") ELSE PUTTEXT("   MOD");      40585000
      END END ELSE                                                             40586000
      IF CURTYPE=BOOLTYPE OR F=SET THEN                                        40587000
      BEGIN                                                                    40588000
        MODE:=BITPATTERN;                                                      40589000
        IF CURMODE≠MODE THEN                                                   40590000
        BEGIN SYMTAB[STARTSYM]:="    B("; PUTSYM(")") END;                     40591000
        PUTTEXT("  AND ");                                                     40592000
        IF CURSY≠(IF F=SET THEN ASTERISK ELSE ANDSY) THEN ERROR(64);           40593000
      END ELSE ERROR(64);                                                      40594000
```

```
      PUTDUMMY; STARTSYM:=NUMSYMS;                                          40595000
      INSYMBOL; FACTOR;                                                     40596000
      IF CURTYPE>0 AND TYPE1>0 THEN                                         40597000
      BEGIN                                                                 40598000
        IF CURTYPE≠TYPE1 THEN                                               40599000
        BEGIN                                                               40600000
          IF TYPETAB[TYPE1].FORM≠NUMERIC OR CURTYPE≠REALTYPE THEN           40601000
          CHECKTYPES(TYPE1,CURTYPE);                                        40602000
          IF TYPE1=REALTYPE THEN CURTYPE:=REALTYPE;                         40603000
        END;                                                                40604000
        IF CURTYPE=REALTYPE AND MULOPTR≥DIVSY THEN ERROR(65);               40605000
      END;                                                                  40606000
      IF MULOPTR=SLASH THEN CURTYPE:=REALTYPE;                              40607000
      IF CURTYPE=0 THEN CURTYPE:=TYPE1;                                     40608000
    END OF WHILE LOOP;                                                      40609000
    IF MODE=BITPATTERN AND CURMODE≠MODE THEN                                40610000
    BEGIN SYMTAB[STARTSYM]:="    B("; PUTSYM(")") END;                      40611000
    CURMODE:=MODE;                                                          40612000
END OF TERM;                                                                40613000
                                                                            40614000
                                                                            40615000
PROCEDURE SIMPLEEXPRESSION;              %*** SIMPLE EXPRESSION ***         40616000
BEGIN                                    %************************          40617000
    INTEGER STARTSYM,MODE,TYPE1,F;                                          40618000
    BOOLEAN SIGNED;                                                         40619000
                                                                            40620000
    PUTDUMMY; STARTSYM:=NUMSYMS;                                            40621000
    IF CURSY=PLUS OR CURSY=MINUS THEN                                       40622000
    BEGIN SIGNED:=TRUE;                                                     40623000
      PUTSYM(IF CURSY=PLUS THEN"+" ELSE "-");                               40624000
      INSYMBOL;                                                             40625000
    END;                                                                    40626000
    TERM;                                                                   40627000
    MODE:=CURMODE;                                                          40628000
    IF SIGNED THEN                                                          40629000
    BEGIN F:=TYPETAB[CURTYPE].FORM;                                         40630000
      IF F≠NUMERIC AND F≠FLOATING THEN ERROR(29);                           40631000
    END;                                                                    40632000
    WHILE CURSY>PLUS AND CURSY≤ORSY DO      % "+","-","OR"                  40633000
    BEGIN                                                                   40634000
      TYPE1:=CURTYPE; F:=TYPETAB[TYPE1].FORM;                               40635000
      IF F=NUMERIC OR F=FLOATING THEN                                       40636000
      BEGIN MODE:=NUMBER;                                                   40637000
        IF CURSY=PLUS THEN PUTSYM("+") ELSE                                 40638000
        IF CURSY=MINUS THEN PUTSYM("-") ELSE ERROR(64);                     40639000
      END ELSE                                                              40640000
      IF CURTYPE=BOOLTYPE THEN                                              40641000
      BEGIN                                                                 40642000
        MODE:=BITPATTERN;                                                   40643000
        IF CURMODE≠MODE THEN                                                40644000
        BEGIN SYMTAB[STARTSYM]:="    B("; PUTSYM(")") END;                  40645000
        IF CURSY=ORSY THEN PUTTEXT("    OR") ELSE ERROR(64);                40646000
      END ELSE                                                              40647000
      IF F=SET THEN                                                         40648000
      BEGIN                                                                 40649000
        MODE:=BITPATTERN;                                                   40650000
        IF CURMODE≠MODE THEN                                                40651000
```

```
        BEGIN SYMTAB[STARTSYM]:="    B(") PUTSYM(")") END;                    40652000
        IF CURSY=PLUS THEN PUTTEXT("    OR") ELSE                            40653000
        IF CURSY=MINUS THEN BEGIN PUTTEXT("   AND");PUTTEXT("  NOT ")END    40654000
        ELSE ERROR(64);                                                      40655000
      END ELSE ERROR(64);                                                    40656000
      INSYMBOL;                                                              40657000
      PUTDUMMY;  STARTSYM:=NUMSYMS;                                          40658000
      TERM;                                                                  40659000
      IF CURTYPE>0 AND TYPE1>0 THEN                                          40660000
      BEGIN                                                                  40661000
        IF CURTYPE≠TYPE1 THEN                                                40662000
        BEGIN                                                                40663000
          IF TYPETAB[TYPE1].FORM≠NUMERIC OR CURTYPE≠REALTYPE THEN            40664000
          CHECKTYPES(TYPE1,CURTYPE);                                         40665000
          IF TYPE1=REALTYPE THEN CURTYPE:=REALTYPE;                          40666000
      END END;                                                               40667000
      IF CURTYPE=0 THEN CURTYPE:=TYPE1;                                      40668000
    END OF WHILE LOOP;                                                       40669000
    IF MODE=BITPATTERN AND CURMODE≠BITPATTERN THEN                           40670000
    BEGIN SYMTAB[STARTSYM]:="    B(") PUTSYM(")") END;                       40671000
    CURMODE:=MODE;                                                           40672000
END OF SIMPLEEXPRESSION;                                                     40673000
                                                                             40674000
                                                                             40675000
PROCEDURE EXPRESSION;                    %*** EXPRESSION ***                 40676000
BEGIN                                    %*******************                40677000
    INTEGER STARTSYM,FIRSTSYM,TYPE1,RELOPTR,F;                               40678000
    BOOLEAN CALLGEN;                                                         40679000
                                                                             40680000
EXPRLEVEL:=EXPRLEVEL+1;                                                      40681000
IF EXPRLEVEL = 1 THEN                                                        40682000
BEGIN                                                                        40683000
  PUTDUMMY;                                                                  40684000
  FIRSTSYM := NUMSYMS;                                                       40685000
END;                                                                         40686000
PUTDUMMY;  STARTSYM:=NUMSYMS;                                                40687000
PUTDUMMY;                                                                    40688000
SIMPLEEXPRESSION;                                                            40689000
IF CURSY≥LSSSY AND CURSY≤INSY THEN    % "<","≤","≥",">","=","≠","IN"         40690000
BEGIN                                                                        40691000
  TYPE1:=CURTYPE;  F:=TYPETAB[TYPE1].FORM;                                   40692000
  RELOPTR:=CURSY;                                                            40693000
  IF F≤ALFA THEN                                                             40694000
  BEGIN                                                                      40695000
    IF CURMODE=BITPATTERN THEN                                               40696000
    BEGIN SYMTAB[STARTSYM]:=" REAL(") PUTSYM(")") END;                       40697000
    IF CURSY=LSSSY THEN PUTSYM("<") ELSE                                     40698000
    IF CURSY=LEQSY THEN PUTSYM("≤") ELSE                                     40699000
    IF CURSY=GEQSY THEN PUTSYM("≥") ELSE                                     40700000
    IF CURSY=GTRSY THEN PUTSYM(">") ELSE                                     40701000
    IF CURSY=EQLSY THEN PUTSYM("=") ELSE                                     40702000
    IF CURSY=NEQSY THEN PUTSYM("≠") ELSE                                     40703000
    BEGIN                                                                    40704000
      IF F≥FLOATING THEN ERROR(64);                                          40705000
      SYMTAB[STARTSYM]:="INTST("  PUTSYM(",");  CALLGEN:=TRUE;               40706000
    END;                                                                     40707000
  END ELSE                                                                   40708000
```

```
IF F=SET THEN                                                                  40709000
BEGIN                                                                          40710000
  IF CURMODE=BITPATTERN THEN                                                   40711000
  BEGIN SYMTAB[STARTSYM+1]:=" REAL(" PUTSYM(")") END;                          40712000
  IF CURSY=EQLSY OR CURSY=NEQSY THEN                                           40713000
  BEGIN PUTSYM(IF CURSY=EQLSY THEN "=" ELSE "≠");                              40714000
  END ELSE                                                                     40715000
  BEGIN                                                                        40716000
    IF CURSY=LEQSY THEN SYMTAB[STARTSYM]:="INCL1(" ELSE                        40717000
    IF CURSY=GEQSY THEN SYMTAB[STARTSYM]:="INCL2(" ELSE ERROR(64);             40718000
    PUTSYM(",");  CALLGEN:=TRUE;                                              40719000
END END ELSE                                                                   40720000
IF F=POINTERS THEN                                                             40721000
BEGIN                                                                          40722000
  IF CURSY=EQLSY THEN PUTSYM("=") ELSE                                         40723000
  IF CURSY=NEQSY THEN PUTSYM("≠") ELSE ERROR(64);                              40724000
END ELSE ERROR(64);                                                            40725000
INSYMBOL;                                                                      40726000
PUTDUMMY;  STARTSYM:=NUMSYMS;                                                  40727000
SIMPLEEXPRESSION;                                                              40728000
IF CURTYPE>0 AND TYPE1>0 THEN                                                  40729000
  IF CURTYPE≠TYPE1 THEN                                                        40730000
    IF RELOPTR≠INSY THEN                                                       40731000
    BEGIN                                                                      40732000
      IF TYPETAB[TYPE1].FORM≠NUMERIC OR CURTYPE≠REALTYPE THEN                  40733000
      CHECKTYPES(TYPE1,CURTYPE);                                               40734000
    END ELSE                                                                   40735000
    IF TYPETAB[CURTYPE].FORM≠SET THEN ERROR(66)                                40736000
              ELSE CHECKTYPES(TYPE1,TYPETAB[CURTYPE].SETTYPE);                 40737000
IF CURMODE=BITPATTERN THEN                                                     40738000
BEGIN SYMTAB[STARTSYM]:=" REAL(" PUTSYM(")") END;                              40739000
  IF CALLGEN THEN PUTSYM(";")                                                  40740000
  CURTYPE:=BOOLTYPE;  CURMODE:=BITPATTERN;                                     40741000
END;                                                                           40742000
EXPRLEVEL:=EXPRLEVEL-1;                                                        40743000
IF EXPRLEVEL=0 THEN                                                            40744000
BEGIN                                                                          40745000
  IF CURMODE=BITPATTERN THEN                                                   40746000
  BEGIN                                                                        40747000
    SYMTAB[FIRSTSYM] := " BEAL(";                                              40748000
    PUTSYM(")");                                                               40749000
  END;                                                                         40750000
  WRITEEXPR;                                                                   40751000
END;                                                                           40752000
END OF EXPRESSION;                                                             40753000
                                                                               40754000
                                                                               40755000
DEFINE BOOLEXPR=                                                               40756000
BEGIN                                                                          40757000
    PUTDUMMY;  EXPRLEVEL:=1;  EXPRESSION;                                      40758000
    IF CURTYPE>0 THEN IF CURTYPE≠BOOLTYPE THEN ERROR(17);                      40759000
    IF CURMODE≠BITPATTERN THEN                                                 40760000
    BEGIN SYMTAB[1]:="    B(";  PUTSYM(")") END;                               40761000
    EXPRLEVEL:=0;  WRITEEXPR;                                                  40762000
END OF BOOLEXPR#;                                                              40763000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%50001000
%                                                                              %50002000
```

```
%                                                                                50003000
%                                                                                50004000
%                    PART 5:   INTRINSIC ROUTINES.                               50005000
%                              ------------------                                50006000
%                                                                                50007000
%                                                                                50008000
%                                                                                50009000
%*******************************************************************************50010000
                                                                                 50011000
                                                                                 50012000
PROCEDURE CONCAT;                          %*** "CONCAT" ***                     50013000
BEGIN                                      %*****************                    50014000
    DEFINE INTEXPR=                                                              50015000
    BEGIN INSYMBOL; EXPRESSION;                                                  50016000
          IF CURTYPE>0 THEN                                                      50017000
          IF TYPETAB[CURTYPE].FORM#NUMERIC THEN ERROR(17);                       50018000
    END #;                                                                       50019000
                                                                                 50020000
    PUTTEXT("CONCAT");  PUTSYM("(");                                             50021000
    INSYMBOL;                                                                    50022000
    IF CURSY=LPAR THEN                                                           50023000
    BEGIN                                                                        50024000
      INSYMBOL;  EXPRESSION;                                                     50025000
      IF CURTYPE>0 THEN                                                          50026000
      IF TYPETAB[CURTYPE].FORM>ALFA THEN ERROR(17);                              50027000
      IF CURSY=COMMA THEN                                                        50028000
      BEGIN                                                                      50029000
        PUTSYM(","); INSYMBOL;  EXPRESSION;                                      50030000
        IF CURTYPE>0 THEN                                                        50031000
        IF TYPETAB[CURTYPE].FORM>ALFA THEN ERROR(17);                            50032000
        IF CURSY=COMMA THEN                                                      50033000
        BEGIN                                                                    50034000
          PUTSYM(","); INTEXPR;                                                  50035000
          IF CURSY=COMMA THEN                                                    50036000
          BEGIN                                                                  50037000
            PUTSYM(","); INTEXPR;                                                50038000
            IF CURSY=COMMA THEN                                                  50039000
            BEGIN                                                                50040000
              PUTSYM(","); INTEXPR;                                              50041000
              PUTSYM(","); PUTCONST(CARDCNT);                                    50042000
              PUTSYM(")");                                                       50043000
              IF CURSY#RPAR THEN BEGIN ERROR(3); SKIP(RPAR) END;                 50044000
            END ELSE BEGIN ERROR(3); SKIP(RPAR) END;                             50045000
          END ELSE BEGIN ERROR(3); SKIP(RPAR) END;                               50046000
        END ELSE BEGIN ERROR(3); SKIP(RPAR) END;                                 50047000
      END ELSE BEGIN ERROR(3); SKIP(RPAR) END;                                   50048000
    END ELSE BEGIN ERROR(3); SKIP(RPAR) END;                                     50049000
    CURTYPE:=REALTYPE;                                                           50050000
    IF CURSY=RPAR THEN INSYMBOL;                                                 50051000
END OF CONCAT;                                                                   50052000
                                                                                 50053000
                                                                                 50054000
PROCEDURE PREAD(CHANGELINE);                                                     50055000
VALUE CHANGELINE; BOOLEAN CHANGELINE;                                            50056000
BEGIN                                                                            50057000
    INTEGER FILID,F;                                                             50058000
    BOOLEAN CHECK;                                                               50059000
```

```
GEN(" BEGIN",7,2);
FILEPARAM(INPUTFILE); FILEID:=FILENAME;
IF TYPETAB1[CURTYPE].FORM=FILES THEN ERROR(85);
IF SYMKIND[CURSY]≠TERMINAL THEN
BEGIN
  IF CURSY NEQ RPAR THEN
  DO BEGIN
    WHILE CURSY=COMMA DO INSYMBOL;
    IF CURSY=IDENTIFIER THEN
    BEGIN
      SEARCH;
      IF FOUND THEN
      BEGIN
       IF THISID.IDCLASS=VAR OR
          THISID.IDCLASS=CONST AND BOOLEAN(THISID.FORMAL) THEN
       BEGIN
        VARIABLE; F:=TYPETAB1[CURTYPE].FORM;
        IF F=NUMERIC OR F=FLOATING OR F=CHAR THEN
        BEGIN
          CHECK:=CHECKOPTION AND F≠FLOATING;
          WRITEXPR; GEN(":=",2,6);
          IF CHECK THEN GEN("CHECK(",6,2);
          GEN("PREAD(",6,2); GENID("F",FILEID,5); GEN(",",1,7);
          GENID("V",FILEID,5); GEN(",",1,7);
          GENID("I",FILEID,5); GEN(",",1,7);
          IF F=NUMERIC THEN GENINT(2) ELSE
          IF F=FLOATING THEN GENINT(3) ELSE GENINT(1);
          GEN(",",1,7); GENINT(CARDCNT); GEN(")",1,7);
          IF CHECK THEN
          BEGIN
            GEN(",",1,7); GENINT(TYPETAB2[CURTYPE]); GEN(",",1,7);
            GENINT(TYPETAB3[CURTYPE]); GEN(",",1,7);
            GENINT(CARDCNT); GEN(")",1,7);
          END;
        END ELSE BEGIN ERROR(82); INSYMBOL END;
       END ELSE BEGIN ERROR(8); INSYMBOL END;
      END ELSE BEGIN ERROR(1); INSYMBOL END;
    END ELSE ERROR(9);
    GEN(";",1,7);
  END UNTIL CURSY≠COMMA;
  IF CURSY≠RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;
  IF CURSY=RPAR THEN INSYMBOL;
END;
IF CHANGELINE THEN
BEGIN
  GEN("RLINE(",6,2); GENID("F",FILEID,5); GEN(",",1,7);
  GENID("V",FILEID,5); GEN(",",1,7);
  GENID("I",FILEID,5); GEN(")",1,7);
END;
GEN("END",4,5);
END OF PREAD;


PROCEDURE PWRITE(LINEFEED);
VALUE LINEFEED; BOOLEAN LINEFEED;
BEGIN
  INTEGER FILEID,F,I,LASTSY;
```

```
POINTER P;                                                              50117000
GEN(" BEGIN",7,2);                                                      50118000
FILEPARAM(OUTPUTFILE); FILEID:=FILENAME;                                50119000
IF TYPETAB[CURTYPE].FORM=FILES THEN ERROR(85);                          50120000
IF SYMKIND[CURSY]#TERMINAL THEN                                         50121000
BEGIN                                                                   50122000
  IF CURSY NEQ RPAR THEN                                                50123000
  DO BEGIN                                                              50124000
    WHILE CURSY=COMMA DO INSYMBOL;                                      50125000
    IF CURSY=ALFACONST AND CURLENGTH>7 THEN                             50126000
    BEGIN                                                               50127000
      GEN("WALFA(",6,2); GENID("F",FILEID,5); GEN(",",1,7);             50128000
      GENID("V",FILEID,5); GEN(",",1,7);                                50129000
      GENID("I",FILEID,5); GEN(",",1,7);                                50130000
      P:=STRINGPNT;                                                     50131000
      FOR I:=1 STEP 7 UNTIL 80 DO                                       50132000
      IF I≤CURLENGTH THEN                                               50133000
      BEGIN                                                             50134000
        IF ALGOLCNT<10 THEN WRITEALGOL;                                 50135000
        REPLACE ALGOLPNT:ALGOLPNT BY """, P:P FOR 7, """, ",";          50136000
        ALGOLCNT:=ALGOLCNT+10;                                          50137000
      END ELSE GEN("0,",2,4);                                           50138000
      GENINT(CURLENGTH); GEN(",",1,7);                                  50139000
      GENINT(CARDCNT); GEN(")",1,7);                                    50140000
      INSYMBOL;                                                         50141000
    END OF ALFACONST ELSE                                               50142000
BEGIN                                                                   50143000
  GEN("PWRITE(",7,1); GENID("F",FILEID,5); GEN(",",1,7);                50144000
  GENID("V",FILEID,5); GEN(",",1,7);                                    50145000
  GENID("I",FILEID,5); GEN(",",1,7);                                    50146000
  LASTSY:=CURSY;                                                        50147000
  EXPRESSION; F:=TYPETAB[CURTYPE].FORM;                                 50148000
  GEN(",",1,7);                                                         50149000
  IF F=NUMERIC OR F=FLOATING OR F=CHAR OR F=ALFA OR                     50150000
     CURTYPE=BOOLTYPE THEN                                              50151000
  BEGIN                                                                 50152000
    IF F=NUMERIC  THEN GENINT(1) ELSE                                   50153000
    IF F=FLOATING THEN GENINT(2) ELSE                                   50154000
    IF F=ALFA     THEN GENINT(5) ELSE                                   50155000
    IF F=CHAR     THEN GENINT(4) ELSE GENINT(3);                        50156000
    GEN(",",1,7);                                                       50157000
    IF CURSY=COLON THEN                                                 50158000
    BEGIN                                                               50159000
      INSYMBOL; EXPRESSION;                                             50160000
      IF TYPETAB[CURTYPE].FORM NEQ NUMERIC THEN ERROR(17);              50161000
      GEN(",",1,7);                                                     50162000
      IF CURSY=COLON THEN                                               50163000
      BEGIN                                                             50164000
        IF F#FLOATING THEN ERROR(4);                                    50165000
        INSYMBOL; EXPRESSION;                                           50166000
        IF TYPETAB[CURTYPE].FORM NEQ NUMERIC THEN ERROR(17);            50167000
        GEN(",",1,7);                                                   50168000
      END ELSE GEN("-1.",3,5);                                          50169000
    END ELSE                                                            50170000
    BEGIN                                                               50171000
      IF F=FLOATING THEN GENINT(16) ELSE                                50172000
      IF F=ALFA AND LASTSY=ALFACONST THEN GENINT(CURLENGTH) ELSE        50173000
```

```
                IF F=ALFA     THEN GENINT(7)  ELSE                                50174000
                IF F=CHAR     THEN GENINT(1) ELSE  GENINT(10);                    50175000
                GEN(",-1,",4,4);                                                  50176000
              END;                                                                50177000
            END ELSE ERROR(17);                                                   50178000
            GENINT(CARDCNT);  GEN(")",1,7);                                       50179000
          END OF EXPRESSION;                                                      50180000
          GEN("}",1,7);                                                           50181000
        END UNTIL CURSY#COMMA;                                                    50182000
        IF CURSY#RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;                       50183000
        IF CURSY=RPAR THEN INSYMBOL;                                              50184000
      END;                                                                        50185000
      FILENAME:=FILEID;                                                           50186000
      IF LINEFEED THEN                                                            50187000
      BEGIN                                                                       50188000
        INTEGER DUMMY;                                                            50189000
        GEN("WLINE(",6,2);  GENID("F",FILENAME,5);  GEN(",",1,7);                 50190000
        GENID("V",FILENAME,5);  GEN(",",1,7);                                     50191000
        GENID("I",FILENAME,5);  GEN(")",1,7);                                     50192000
      END;                                                                        50193000
      GEN("END",4,5);                                                             50194000
END OF PWRITE;                                                                    50195000
                                                                                  50196000
                                                                                  50197000
PROCEDURE FILEHANDLING(PROCNUM);          %*** FILE HANDLING PROCEDURES;          50198000
VALUE PROCNUM; INTEGER PROCNUM;           %***                                    50199000
BEGIN                                     %***    1) PUT                          50200000
    INTEGER F;                            %***    2) GET                          50201000
    CASE PROCNUM OF                       %***    3) RESET                        50202000
    BEGIN ;                               %***    4) REWRITE                      50203000
      GEN("PUT",3,5);                     %***    5) PAGE                         50204000
      GEN("GET",3,5); %                                                           50205000
      GEN("RESET",5,3); %                                                         50206000
      GEN("REWRITE",7,1); %                                                       50207000
      GEN("PAGE",4,4); %                                                          50208000
    END; %                                                                        50209000
    GEN("(",1,7);  FILEPARAM(0); %                                                50210000
    IF FILENAME=0 THEN ERROR(78); %                                               50211000
    F:=TYPETAB[CURTYPE].FORM;                                                     50212000
    IF F=FILES AND PROCNUM=5 THEN ERROR(80);                                      50213000
    GENID("F",FILENAME,5);  GEN(",",1,7);                                         50214000
    GENID("V",FILENAME,5);  GEN(",",1,7);                                         50215000
    GENID("I",FILENAME,5);  GEN(",",1,7);                                         50216000
    GENINT(CARDCNT);  GEN(")",1,7);                                               50217000
    IF CURSY#RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;                           50218000
    IF CURSY=RPAR THEN INSYMBOL;                                                  50219000
END OF FILEHANDLING;                                                              50220000
                                                                                  50221000
                                                                                  50222000
PROCEDURE PACK;                                                                   50223000
BEGIN                                                                             50224000
    INTEGER IT,T;                                                                 50225000
    GEN("PACK(",5,3);                                                             50226000
    INSYMBOL;                                                                     50227000
    IF CURSY=LPAR THEN                                                            50228000
    BEGIN                                                                         50229000
      INSYMBOL;                                                                   50230000
```

```
IF CURSY=IDENTIFIER THEN                                                    50231000
BEGIN                                                                       50232000
  SEARCH;                                                                   50233000
  IF FOUND THEN                                                             50234000
  BEGIN                                                                     50235000
    IF THISID.IDCLASS=VAR THEN                                              50236000
    BEGIN                                                                   50237000
      T:=TYPETAB1[THISID.TYPE];                                             50238000
      IF T.FORM=ARRAYS THEN                                                 50239000
      BEGIN                                                                 50240000
        IT:=T.INXTYPE;                                                      50241000
        IF TYPETAB1[T.ARRTYPE].FORM#CHAR THEN ERROR(88);                    50242000
        GENID("V",1000*THISLEVEL+THISINDEX,5);                              50243000
        IF THISLEVEL>1 AND THISLEVEL#CURLEVEL THEN ERROR(5);                50244000
        GEN(",",1,7);  GENINT(TYPETAB2[THISID.TYPE]);                       50245000
        GEN(",",1,7);  GENINT(TYPETAB3[THISID.TYPE]);                       50246000
      END ELSE ERROR(88);                                                   50247000
    END ELSE ERROR(88);                                                     50248000
  END ELSE ERROR(1);                                                        50249000
END ELSE ERROR(9);                                                          50250000
INSYMBOL;                                                                   50251000
IF CURSY=COMMA THEN                                                         50252000
BEGIN                                                                       50253000
  GEN(",",1,7);                                                             50254000
  INSYMBOL;  EXPRESSION;  CHECKTYPES(IT,CURTYPE);                           50255000
  IF CURSY=COMMA THEN                                                       50256000
  BEGIN                                                                     50257000
    GEN(",",1,7);                                                           50258000
    INSYMBOL;                                                               50259000
    IF CURSY=IDENTIFIER THEN                                                50260000
    BEGIN                                                                   50261000
      SEARCH;                                                               50262000
      IF FOUND THEN                                                         50263000
      BEGIN                                                                 50264000
        IF THISID.IDCLASS=VAR OR                                            50265000
           THISID.IDCLASS=CONST AND BOOLEAN(THISID.FORMAL) THEN             50266000
        BEGIN                                                               50267000
          VARIABLE;  WRITEEXPR;                                             50268000
          IF CURTYPE>0 THEN                                                 50269000
          IF TYPETAB1[CURTYPE].FORM#ALFA THEN ERROR(12);                    50270000
        END ELSE ERROR(8);                                                  50271000
      END ELSE ERROR(1);                                                    50272000
    END ELSE ERROR(9);                                                      50273000
  END ELSE BEGIN ERROR(89); SKIP(RPAR) END;                                 50274000
  END ELSE BEGIN ERROR(89); SKIP(RPAR) END;                                 50275000
  IF CURSY#RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;                       50276000
  IF CURSY=RPAR THEN INSYMBOL;                                              50277000
END ELSE BEGIN ERROR(3); INSYMBOL END;                                      50278000
GEN(",",1,7);  GENINT(CARDCNT);  GEN(")",1,7);                              50279000
END OF PACK;                                                                50280000
                                                                            50281000
                                                                            50282000
PROCEDURE UNPACK;                                                           50283000
BEGIN                                                                       50284000
  INTEGER IT,T;                                                             50285000
  GEN("UNPACK(",7,1);  INSYMBOL;                                            50286000
  IF CURSY=LPAR THEN                                                        50287000
```

```
BEGIN                                                                   50288000
  INSYMBOL; EXPRESSION;                                                 50289000
  IF CURTYPE>0 THEN IF TYPETAB1[CURTYPE].FORM#ALFA THEN ERROR(17);      50290000
  IF CURSY=COMMA THEN                                                   50291000
  BEGIN                                                                 50292000
    GEN(",",1,7);  INSYMBOL;                                            50293000
    IF CURSY=IDENTIFIER THEN                                            50294000
    BEGIN                                                               50295000
      SEARCH;                                                           50296000
      IF FOUND THEN                                                     50297000
      BEGIN                                                             50298000
        IF THISID.IDCLASS=VAR THEN                                      50299000
        BEGIN                                                           50300000
          T:=TYPETAB1[THISID.TYPE];                                     50301000
          IF T.FORM=ARRAYS THEN                                         50302000
          BEGIN                                                         50303000
            IT:=T.INXTYPE;                                              50304000
            IF TYPETAB1[T.ARRTYPE].FORM#CHAR THEN ERROR(88);            50305000
            IF THISLEVEL>1 AND THISLEVEL#CURLEVEL THEN ERROR(5);        50306000
            GENID("V",1000*THISLEVEL+THISINDEX,5);                      50307000
            GEN(",",1,7);  GENINT(TYPETAB2[THISID.TYPE]);               50308000
            GEN(",",1,7);  GENINT(TYPETAB3[THISID.TYPE]);               50309000
          END ELSE ERROR(88);                                           50310000
        END ELSE ERROR(88);                                             50311000
      END ELSE ERROR(1);                                                50312000
    END ELSE ERROR(9);                                                  50313000
    INSYMBOL;                                                           50314000
    IF CURSY=COMMA THEN                                                 50315000
    BEGIN                                                               50316000
      GEN(",",1,7);                                                     50317000
      INSYMBOL; EXPRESSION; CHECKTYPES(IT,CURTYPE);                     50318000
    END ELSE BEGIN ERROR(80); SKIP(RPAR) END;                           50319000
  END ELSE BEGIN ERROR(89); SKIP(RPAR) END;                             50320000
  IF CURSY#RPAR THEN BEGIN ERROR(89); SKIP(RPAR) END;                   50321000
  IF CURSY=RPAR THEN INSYMBOL;                                          50322000
END ELSE BEGIN ERROR(3); INSYMBOL END;                                  50323000
GEN(",",1,7);  GENINT(CARDCNT);  GEN(")",1,7);                          50324000
END OF UNPACK;                                                          50325000
                                                                        50326000
                                                                        50327000
PROCEDURE NEWDISP;                          %*** "NEW","DISPOSE"        50328000
BEGIN                                                                   50329000
  INTEGER T1;                                                           50330000
  IF CURNAME1="3000NEW" THEN GEN("NEW(",4,4) ELSE                       50331000
  BEGIN GEN("DISPOSE",7,1); GEN("(",1,7) END;                           50332000
  INSYMBOL;                                                             50333000
  IF CURSY=LPAR THEN                                                    50334000
  BEGIN                                                                 50335000
    INSYMBOL;                                                           50336000
    IF CURSY=IDENTIFIER THEN                                            50337000
    BEGIN                                                               50338000
      SEARCH;                                                           50339000
      IF FOUND THEN                                                     50340000
      BEGIN                                                             50341000
        VARIABLE;                                                       50342000
        IF CURTYPE>0 THEN IF TYPETAB1[CURTYPE].FORM=POINTERS THEN       50343000
        BEGIN                                                           50344000
```

```
            WRITEEXPR; GEN(",",1,7);                                          50345000
            T1:=TYPETAB1[CURTYPE].POINTTYPE;                                  50346000
            T1:=TYPETAB1[T1].SIZE;                                            50347000
            IF T1>1023 THEN ERROR(86);                                        50348000
            GENINT(T1); GEN("I",1,7);                                         50349000
          END ELSE ERROR(81);                                                 50350000
        END ELSE BEGIN ERROR(1); INSYMBOL END;                                50351000
      END ELSE ERROR(9);                                                      50352000
      WHILE CURSY=COMMA DO                                                    50353000
      BEGIN INSYMBOL;                                                         50354000
            IF CURSY NEQ IDENTIFIER THEN ERROR(9);                            50355000
            IF CURSY NEQ RPAR THEN INSYMBOL;                                  50356000
      END;                                                                    50357000
    END ELSE BEGIN ERROR(58); SKIP(RPAR) END;                                 50358000
    IF CURSY#RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;                       50359000
    IF CURSY=RPAR THEN INSYMBOL;                                              50360000
END OF NEWDISP;                                                               50361000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%60001000
%                                                                             %60002000
%                                                                             %60003000
%                                                                             %60004000
%                  PART 6:  THE STATEMENT PARSER.                             %60005000
%                           ---------------------                             %60006000
%                                                                             %60007000
%                                                                             %60008000
%                                                                             %60009000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%60010000
                                                                              60011000
                                                                              60012000
                                                                              60013000
PROCEDURE STATEMENT;      FORWARD;                                            60014000
                                                                              60015000
PROCEDURE ASSIGNMENT;                                                         60016000
BEGIN                                                                         60017000
    INTEGER LEFTTYPE;                                                         60018000
    LABEL ASSIGN,EXIT;                                                        60019000
    IF FOUND THEN                                                             60050000
    BEGIN                                                                     60051000
      IF THISID.IDCLASS=VAR OR                                                60052000
         THISID.IDCLASS=CONST AND BOOLEAN(THISID.FORMAL) THEN                 60053000
      BEGIN                                                                   60054000
        VARIABLE; LEFTTYPE:=CURTYPE;                                          60055000
ASSIGN: IF CURSY#ASSIGNSY THEN                                                60056000
        BEGIN ERROR(28); SKIP(ASSIGNSY);                                      60057000
              IF SYMKIND[CURSY]=TERMINAL THEN GO TO EXIT;                     60058000
        END;                                                                  60059000
        INSYMBOL;                                                             60060000
        IF TYPETAB1[LEFTTYPE].STRUCT>0 THEN                                   60061000
        BEGIN                                                                 60062000
          ERROR(95);                                                          60063000
        END ELSE                                                              60080000
        BEGIN                                                                 60081000
          WRITEFXPR; GEN(":=",2,6);                                           60082000
          IF CHECKOPTION AND TYPETAB1[LEFTTYPE].FORM#CHAR THEN                60083000
          CHECKEXPR(TYPETAB2[LEFTTYPE],TYPETAB3[LEFTTYPE]) ELSE               60084000
          EXPRESSION;                                                         60085000
          WRITEEXPR;                                                          60086000
```

```
          CHECKTYPES(LEFTTYPE,CURTYPE);                                          60087000
        END;                                                                     60088000
      END ELSE                                                                   60089000
      BEGIN  % FUNCTION ASSIGNMENT.                                              60090000
        IF THISLEVEL≠CURLEVEL-1 OR THISINDEX≠CURFUNC THEN ERROR(5);              60091000
        GENID("V",1000*THISLEVEL+THISINDEX,5);  LEFTTYPE:=THISID.TYPE;           60092000
        INSYMBOL;  GO TO ASSIGN;                                                 60093000
      END;                                                                       60094000
    END ELSE                                                                     60095000
    BEGIN                                                                        60096000
      SKIP(ASSIGNSY);                                                            60097000
      IF CURSY=ASSIGNSY THEN GO TO ASSIGN;                                       60098000
    END;                                                                         60099000
EXIT:                                                                            60100000
END OF ASSIGNMENT;                                                               60101000
                                                                                 60102000
                                                                                 60103000
PROCEDURE COMPSTAT;                                                              60104000
BEGIN                                                                            60105000
    INTEGER BEGINNUM;                                                            60106000
    LABEL STATM;                                                                 60107000
                                                                                 60108000
    BEGINNUM:=NUMBEGINS:=NUMBEGINS+1;  MARGIN(" B",BEGINNUM);                    60109000
    GEN("BEGIN",6,3);                                                            60110000
    DO BEGIN                                                                     60111000
        IF CURSY=SEMICOLON OR CURSY=BEGINSY THEN INSYMBOL;                       60112000
STATM:  STATEMENT;                                                               60113000
        GEN(";",1,7);                                                            60114000
        IF CURSY=ELSESY THEN BEGIN ERROR(20); INSYMBOL; GO STATM END;            60115000
        IF SYMKIND[CURSY]=INITIAL THEN BEGIN ERROR(21); GO STATM END;            60116000
    END UNTIL CURSY≠SEMICOLON;                                                   60117000
    IF CURSY≠ENDSY THEN                                                          60118000
    BEGIN ERROR(24);  SKIP(ENDSY);                                               60119000
        IF CURSY≠ENDSY THEN BEGIN INSYMBOL; GO TO STATM END;                     60120000
    END;                                                                         60121000
    GEN(" END",5,4);  MARGIN(" E",BEGINNUM);                                     60122000
    INSYMBOL;                                                                    60123000
END OF COMPSTAT;                                                                 60124000
                                                                                 60125000
                                                                                 60126000
PROCEDURE IFSTAT;                                                                60127000
BEGIN                                                                            60128000
    LABEL EXIT;                                                                  60129000
    GEN("IF",3,6);                                                               60130000
    INSYMBOL; BOOLEXPR;                                                          60131000
    IF CURSY≠THENSY THEN                                                         60132000
    BEGIN IF CURTYPE>0 THEN ERROR(27);                                           60133000
        SKIP(THENSY);                                                            60134000
        IF CURSY≠THENSY THEN                                                     60135000
        BEGIN IF CURTYPE=0 THEN ERROR(27);                                       60136000
            IF SYMKIND[CURSY]=TERMINAL THEN GO TO EXIT;                          60137000
    END;  END;                                                                   60138000
    GEN(" THEN",6,3);                                                            60139000
    INSYMBOL;  STATEMENT;                                                        60140000
    IF CURSY=ELSESY THEN                                                         60141000
    BEGIN GEN(" ELSE",6,3); INSYMBOL; STATEMENT END;                             60142000
EXIT:                                                                            60143000
```

```
END OF IFSTAT;                                                                  60144000
                                                                                60145000
                                                                                60146000
PROCEDURE CASESTAT;                                                             60147000
BEGIN                                                                           60148000
    DEFINE CASEHASH(N)=(N).[38:39] MOD MAXCASES#;                               60149000
    INTEGER ARRAY CASETAB[0:MAXCASES];                                          60150000
    INTEGER CASENUM,CASETYPE,NCASELABS,TEMPVARNUM,CONVAL,CONTYPE,C,T;           60151000
    BOOLEAN ZEROLAB,FIRST;                                                      60152000
                                                                                60153000
    CASENUM:=NUMCASES:=NUMCASES+1;  MARGIN("CB",CASENUM);                       60154000
    TEMPVARNUM:=NUMTEMPS:=NUMTEMPS+1;                                           60155000
    IF TEMPVARNUM>MAXTEMPS THEN ERROR(16);                                      60156000
    GEN("BEGIN",6,3);  GENID("T",TEMPVARNUM,2);  GEN(":=",2,6);                 60157000
    INSYMBOL;  EXPRESSION;                                                      60158000
    GEN(";",1,7);  CASETYPE:=CURTYPE;                                           60159000
    IF TYPETAB[CASETYPE].FORM>FLOATING THEN                                     60160000
    BEGIN ERROR(17); CASETYPE:=0 END;                                           60161000
    IF CURSY#OFSY THEN                                                          60162000
    BEGIN IF CASETYPE>0 THEN ERROR(18);                                         60163000
          SKIP(OFSY);                                                           60164000
          IF CURSY=OFSY THEN INSYMBOL ELSE                                      60165000
          IF CASETYPE=0 THEN ERROR(18);                                         60166000
    END ELSE INSYMBOL;                                                          60167000
    DO BEGIN                                                                    60168000
      WHILE CURSY=SEMICOLON DO INSYMBOL;                                        60169000
      FIRST:=TRUE;                                                              60170000
      IF CURSY#ENDSY THEN                                                       60171000
BEGIN                                                                           60172000
  GEN("IF",3,6);                                                                60173000
  DO BEGIN                                                                      60174000
  IF FIRST THEN FIRST:=FALSE ELSE INSYMBOL;                                     60175000
    CONSTANT(CONVAL,CONTYPE);                                                   60176000
    IF CONTYPE>0 THEN                                                           60177000
    BEGIN                                                                       60178000
      IF CASETYPE=0 THEN CASETYPE:=CONTYPE ELSE                                 60179000
      CHECKTYPES(CASETYPE,CONTYPE);                                             60180000
      GENID("T",TEMPVARNUM,2);  GEN("=",1,7);  GENINT(CONVAL);                  60181000
      NCASELABS:=NCASELABS+1;                                                   60182000
      IF NCASELABS<MAXCASES THEN                                                60183000
      BEGIN                                                                     60184000
        IF CONVAL=0 THEN                                                        60185000
          IF ZEROLAB THEN ERROR(31) ELSE ZEROLAB:=TRUE   ELSE                   60186000
        BEGIN                                                                   60187000
          T:=CASEHASH(CONVAL);                                                  60188000
          FOR C:=CASETAB[T] WHILE C#CONVAL AND C#0 DO                           60189000
          T:=IF T=0 THEN MAXCASES ELSE T-1;                                     60190000
          IF C#0 THEN ERROR(31) ELSE CASETAB[T]:=CONVAL;                        60191000
        END;                                                                    60192000
      END ELSE IF NCASELABS=MAXCASES THEN ERROR(30);                            60193000
      IF CURSY=COMMA THEN GEN(" OR",4,5);                                       60194000
    END;                                                                        60195000
  END UNTIL CURSY#COMMA;                                                        60196000
  GEN(" THEN",6,3);                                                             60197000
  IF CURSY#COLON THEN BEGIN ERROR(26); SKIP(COLON) END;                         60198000
  IF CURSY=COLON THEN INSYMBOL;                                                 60199000
  STATEMENT;                                                                    60200000
```

```
        IF CURSY≠SEMICOLON AND CURSY≠ENDSY THEN                                  60201000
        BEGIN ERROR(21); SKIP(SEMICOLON) END;                                    60202000
      END;                                                                       60203000
      IF CURSY=SEMICOLON THEN GEN(" ELSE",6,3);                                  60204000
    END UNTIL CURSY≠SEMICOLON;                                                   60205000
    IF CURSY≠ENDSY THEN BEGIN ERROR(24); SKIP(ENDSY) END;                        60206000
    GEN(" END",5,4);  MARGIN("CE",CASENUM);                                      60207000
    NUMTEMPS:=NUMTEMPS-1;                                                        60208000
    INSYMBOL;                                                                    60209000
END OF CASESTAT;                                                                 60210000
                                                                                 60211000
                                                                                 60212000
PROCEDURE WHILESTAT;                                                             60213000
BEGIN                                                                            60214000
    LABEL STATM,EXIT;                                                            60215000
    GEN("WHILE",6,3);                                                            60216000
    INSYMBOL;  BOOLEXPR;                                                         60217000
    IF CURSY≠DOSY THEN                                                           60218000
    BEGIN IF CURTYPE>0 THEN ERROR(19);                                           60219000
          SKIP(DOSY);                                                            60220000
          IF CURSY≠DOSY THEN                                                     60221000
          BEGIN IF CURTYPE=0 THEN ERROR(19);                                     60222000
                GO TO IF SYMKIND[CURSY]=INITIAL THEN STATM ELSE EXIT;            60223000
    END;  END;                                                                   60224000
    GEN(" DO",4,5);                                                              60225000
    INSYMBOL;                                                                    60226000
STATM: STATEMENT;                                                                60227000
EXIT:                                                                            60228000
END OF WHILESTAT;                                                                60229000
                                                                                 60230000
                                                                                 60231000
PROCEDURE REPEATSTAT;                                                            60232000
BEGIN                                                                            60233000
    INTEGER REPNUM;                                                              60234000
    LABEL NEWTRY;                                                                60235000
                                                                                 60236000
    REPNUM:=NUMREPS:=NUMREPS+1;                                                  60237000
    MARGIN(" R",REPNUM);                                                         60238000
    GEN("DO",3,6);  GEN("BEGIN",6,3);                                            60239000
    DO BEGIN                                                                     60240000
        INSYMBOL;                                                                60241000
NEWTRY: STATEMENT;                                                               60242000
        GEN("}",1,7);                                                            60243000
        IF CURSY=ELSESY THEN BEGIN ERROR(20);INSYMBOL; GO NEWTRY END;            60244000
        IF SYMKIND[CURSY]=INITIAL THEN BEGIN ERROR(21); GO NEWTRY END;           60245000
    END UNTIL CURSY≠SEMICOLON;                                                   60246000
    IF CURSY≠UNTILSY THEN                                                        60247000
    BEGIN                                                                        60248000
      ERROR(22);                                                                 60249000
      WHILE CURSY≠UNTILSY AND SYMKIND[CURSY]≠INITIAL DO                          60250000
      BEGIN INSYMBOL; SKIP(UNTILSY) END;                                         60251000
      IF CURSY≠UNTILSY THEN GO TO NEWTRY;                                        60252000
    END;                                                                         60253000
    GEN(" END",5,4);  GEN("UNTIL",6,3);  MARGIN(" U",REPNUM);                    60254000
    INSYMBOL;  BOOLEXPR;                                                         60255000
END OF REPEATSTAT;                                                               60256000
                                                                                 60257000
```

```
                                                                                60258000
PROCEDURE FORSTAT;                                                              60259000
BEGIN                                                                           60260000
    INTEGER VARTYPE,VARNUM,LLIM,ULIM;                                           60261000
    BOOLEAN DOWN;                                                               60262000
    LABEL STATM;                                                                60263000
                                                                                60264000
    GEN("BEGIN",6,3);                                                           60265000
    INSYMBOL;                                                                   60266000
    IF CURSY=IDENTIFIER THEN                                                    60267000
    BEGIN                                                                       60268000
      SEARCH;                                                                   60269000
      IF FOUND THEN                                                             60270000
      BEGIN                                                                     60271000
        VARNUM:=1000*THISLEVEL+THISINDEX;                                       60272000
        IF THISID.IDCLASS=VAR OR                                                60273000
           THISID.IDCLASS=CONST AND BOOLEAN(THISID.FORMAL) THEN                 60274000
        BEGIN                                                                   60275000
          IF THISLEVEL>1 AND THISLEVEL<CURLEVEL THEN ERROR(5);                  60276000
          IF THISLEVEL>CURLEVEL THEN ERROR(83);                                 60277000
          VARTYPE:=THISID.TYPE;                                                 60278000
          IF TYPETAB1[VARTYPE]=FORMSCHAR THEN                                   60279000
          BEGIN                                                                 60280000
            LLIM:=TYPETAB2[VARTYPE]; ULIM:=TYPETAB3[VARTYPE];                   60281000
          END ELSE BEGIN ERROR(12); VARTYPE:=0 END;                             60282000
        END ELSE ERROR(8);                                                      60283000
      END ELSE ERROR(1);                                                        60284000
    END ELSE ERROR(9);                                                          60285000
    INSYMBOL;                                                                   60286000
    IF CURSY#ASSIGNSY THEN                                                      60287000
    BEGIN ERROR(28);                                                            60288000
          SKIP(ASSIGNSY);                                                       60289000
          IF CURSY=ASSIGNSY THEN INSYMBOL ELSE                                  60290000
          IF SYMKIND[CURSY]=INITIAL THEN GO TO STATM;                           60291000
    END ELSE INSYMBOL;                                                          60292000
    GENID("V",VARNUM,5); GEN("+",1,7);                                          60293000
    IF CHECKOPTION THEN CHECKEXPR(LLIM,ULIM) ELSE EXPRESSION;                   60294000
    WRITEEXPR;                                                                  60295000
    GEN("]",1,7);                                                               60296000
    IF VARTYPE=0 THEN VARTYPE:=CURTYPE ELSE CHECKTYPES(VARTYPE,CURTYPE);        60297000
    NUMTEMPS:=NUMTEMPS+1; IF NUMTEMPS>MAXTEMPS THEN ERROR(16);                  60298000
    IF CURSY=TOSY THEN INSYMBOL ELSE                                            60299000
    IF CURSY=DOWNTOSY THEN BEGIN DOWN:=TRUE; INSYMBOL END ELSE                  60300000
    BEGIN IF CURTYPE>0 THEN ERROR(23);                                          60301000
          SKIP(TOSY);                                                           60302000
          IF CURSY=TOSY THEN INSYMBOL ELSE                                      60303000
          BEGIN IF CURTYPE=0 THEN ERROR(23);                                    60304000
                IF SYMKIND[CURSY]=INITIAL THEN GO TO STATM;                     60305000
    END; END;                                                                   60306000
    GENID("T",NUMTEMPS,2); GEN("+",1,7);                                       60307000
    IF CHECKOPTION THEN CHECKEXPR(LLIM,ULIM) ELSE EXPRESSION;                   60308000
    WRITEEXPR;                                                                  60309000
    GEN("]",1,7);                                                               60310000
    IF VARTYPE=0 THEN VARTYPE:=CURTYPE ELSE CHECKTYPES(VARTYPE,CURTYPE);        60311000
    IF CURSY#DOSY THEN                                                          60312000
    BEGIN IF CURTYPE>0 THEN ERROR(19);                                          60313000
          SKIP(DOSY);                                                           60314000
```

```
            IF CURSY=DOSY THEN INSYMBOL ELSE                                    60315000
            IF CURTYPE=0 THEN ERROR(19);                                        60316000
        END ELSE INSYMBOL;                                                      60317000
        GEN("FOR",4,5);  GENID("V",VARNUM,5);  GEN("←",1,7);                    60318000
        GENID("V",VARNUM,5);  GEN(" ",1,7);                                     60319000
        IF DOWN THEN GEN("DOWNTO",7,2) ELSE GEN("UPTO",5,4);                    60320000
        GENID("T",NUMTEMPS,2);  GEN(" DO",4,5);                                 60321000
    STATM: STATEMENT;                                                           60322000
        GEN(" END",5,4);                                                        60323000
        NUMTEMPS:=NUMTEMPS-1;                                                   60324000
    END OF FORSTAT;                                                             60325000
                                                                                60326000
                                                                                60327000
    PROCEDURE GOTOSTAT;                                                         60328000
    BEGIN                                                                       60329000
        INTEGER I;                                                              60330000
        INSYMBOL;                                                               60331000
        IF CURSY=INTCONST THEN                                                  60332000
        BEGIN I:=NUMLABS;                                                       60333000
            WHILE I≥1 AND LABTAB[I].LABVAL≠CURVAL DO I:=I-1;                    60334000
            IF I=0 THEN ERROR(15);                                              60335000
            GEN("GO",3,6);  GENID("L",CURVAL,4);                                60336000
            INSYMBOL;                                                           60337000
        END ELSE ERROR(10);                                                     60338000
    END OF GOTOSTAT;                                                            60339000
                                                                                60340000
                                                                                60341000
    PROCEDURE WITHSTAT;                                                         60342000
    BEGIN                                                                       60343000
        INTEGER STARTLEVEL,VERYFIRSTWITHSYM,I;                                  60344000
        REAL D;                                                                 60345000
        STARTLEVEL:=TOPLEVEL;  VERYFIRSTWITHSYM:=NWITHSYMS;                     60346000
        DO BEGIN                                                                60347000
          INSYMBOL;                                                             60348000
          IF CURSY=IDENTIFIER THEN                                              60349000
          BEGIN                                                                 60350000
            SEARCH;                                                             60351000
            IF FOUND THEN                                                       60352000
            BEGIN                                                               60353000
              IF THISID.IDCLASS=VAR THEN                                        60354000
              BEGIN                                                             60355000
                VARIABLE;                                                       60356000
                IF CURTYPE>0 THEN                                               60357000
                IF TYPETAB1[CURTYPE].FORM≠RECORD THEN ERROR(98);                60358000
                IF SIMPLEVARIABLE THEN                                          60359000
                BEGIN PUTSYM("[");  INSIDEBRACKETS:=TRUE END;                   60360000
                IF TOPLEVEL<MAXLEVEL THEN                                       60361000
                BEGIN                                                           60362000
                  TOPLEVEL:=TOPLEVEL+1;                                         60363000
                  D.NAMETAB:=TYPETAB1[CURTYPE].RECTAB;                          60364000
                  D.RECTYPE:=CURTYPE;                                           60365000
                  D.NUMPNTRSINWITH:=NUMPOINTERS;                                60366000
                  D.FIRSTWITHSYM:=NWITHSYMS;                                    60367000
                  D.BRACKETSINWITH:=REAL(INSIDEBRACKETS);                       60368000
                  IF NWITHSYMS+NUMSYMS>MAXWITHSYMS THEN ERROR(63) ELSE          60369000
                  FOR I:=1 STEP 1 UNTIL NUMSYMS DO                              60370000
                  BEGIN                                                         60371000
```

```
                WITHTAB[NWITHSYMS]:=SYMTAB[I];                                  60372000
                NWITHSYMS:=NWITHSYMS+1;                                         60373000
              END;                                                              60374000
              D.LASTWITHSYM:=NWITHSYMS-1;                                       60375000
              DISPLAY[TOPLEVEL]:=D;                                             60376000
            END ELSE ERROR(84);                                                 60377000
          END ELSE BEGIN ERROR(8); INSYMBOL END;                                60378000
        END ELSE BEGIN ERROR(1); INSYMBOL END;                                  60379000
      END ELSE BEGIN ERROR(9); INSYMBOL END;                                    60380000
    NUMSYMS:=0;                                                                 60381000
    NUMPOINTERS := 0;                                                           60382000
    END UNTIL CURSY#COMMA;                                                      60383000
    IF CURSY#DOSY THEN                                                          60384000
    BEGIN ERROR(19); SKIP(DOSY);                                                60385000
          IF CURSY=DOSY THEN INSYMBOL;                                          60386000
    END ELSE INSYMBOL;                                                          60387000
    STATEMENT;                                                                  60388000
    TOPLEVEL:=STARTLEVEL; NWITHSYMS:=VERYFIRSTWITHSYM;                          60389000
END OF WITHSTAT;                                                                60390000
                                                                                60391000
                                                                                60392000
PROCEDURE STATEMENT;                                                            60393000
BEGIN                                                                           60394000
    INTEGER I;                                                                  60395000
    LABEL LABFOUND;                                                             60396000
                                                                                60397000
    IF CURSY=INTCONST THEN              % *** LABELED STATEMENT ***             60398000
    BEGIN                                                                       60399000
      FOR I:=FIRSTLAB STEP 1 UNTIL NUMLABS DO                                   60400000
      IF LABTAB[I].LABVAL=CURVAL THEN                                           60401000
      BEGIN IF LABTAB[I].LABDEF=1 THEN ERROR(31);                               60402000
            LABTAB[I].LABDEF:=1;                                                60403000
            GO TO LABFOUND;                                                     60404000
      END;                                                                      60405000
      ERROR(15);                                                                60406000
LABFOUND: GENID("L",CURVAL,4); GEN(":",1,7);                                    60407000
      INSYMBOL;                                                                 60408000
      IF CURSY#COLON THEN                                                       60409000
      BEGIN ERROR(26);                                                          60410000
            SKIP(COLON); IF CURSY=COLON THEN INSYMBOL;                          60411000
      END ELSE INSYMBOL;                                                        60412000
    END;                                                                        60413000
                                                                                60414000
    COMMENT *** START OF STATEMENT *** ;                                        60415000
                                                                                60416000
    IF CURSY=IDENTIFIER THEN                                                    60417000
    BEGIN                                                                       60418000
      SEARCH;                                                                   60419000
      IF FOUND THEN                                                             60420000
      BEGIN                                                                     60421000
        IF THISID.IDCLASS=VAR OR                                                60422000
          THISID.IDCLASS=CONST AND BOOLEAN(THISID.FORMAL) OR                    60423000
           THISID.IDCLASS=FUNC THEN ASSIGNMENT ELSE                             60424000
        IF THISID.IDCLASS=PROC THEN                                             60425000
        BEGIN                                                                   60426000
          IF THISLEVEL=0 THEN           % *** INTRINSIC PROCEDURE ***           60427000
          BEGIN                                                                 60428000
```

```
          IF CURNAME1="5OWRITE" THEN PWRITE(FALSE)    ELSE                60429000
          IF CURNAME1="7WRITEL" AND                                       60430000
             CURNAME2="OOOOOON" THEN PWRITE(TRUE)     ELSE                60431000
          IF CURNAME1="4OOREAD" THEN PREAD(FALSE)     ELSE                60432000
          IF CURNAME1="6READLN" THEN PREAD(TRUE)      ELSE                60433000
          IF CURNAME1="4OOPAGE" THEN FILEHANDLING(5)  ELSE                60434000
          IF CURNAME1="3OOOGET" THEN FILEHANDLING(2)  ELSE                60435000
          IF CURNAME1="3OOOPUT" THEN FILEHANDLING(1)  ELSE                60436000
          IF CURNAME1="5ORESET" THEN FILEHANDLING(3)  ELSE                60437000
          IF CURNAME1="7REWRIT" AND                                       60438000
             CURNAME2="OOOOOOF" THEN FILEHANDLING(4)  ELSE                60439000
          IF CURNAME1="3OOONEW" THEN NEWDISP          ELSE                60440000
          IF CURNAME1="7DISPOS" AND                                       60441000
             CURNAME2="OOOOOOF" THEN NEWDISP          ELSE                60442000
          IF CURNAME1="4OOPACK" THEN PACK             ELSE                60443000
          IF CURNAME1="6UNPACK" THEN UNPACK           ELSE ERROR(0)};     60444000
        END ELSE PASSPARAMS};                                             60445000
        WRITEEXPR};                                                       60446000
      END ELSE BEGIN ERROR(1?)}; SKIP(99) END};                           60447000
    END ELSE BEGIN ERROR(1)}; ASSIGNMENT END};                            60448000
END OF IDENTIFIER  ELSE                                                   60449000
IF CURSY=BEGINSY   THEN COMPSTAT    ELSE                                  60450000
IF CURSY=IFSY      THEN IFSTAT      ELSE                                  60451000
IF CURSY=CASESY    THEN CASESTAT    ELSE                                  60452000
IF CURSY=WHILESY   THEN WHILESTAT   ELSE                                  60453000
IF CURSY=REPEATSY  THEN REPEATSTAT  ELSE                                  60454000
IF CURSY=FORSY     THEN FORSTAT     ELSE                                  60455000
IF CURSY=WITHSY    THEN WITHSTAT    ELSE                                  60456000
IF CURSY=GOTOSY    THEN GOTOSTAT    ELSE                                  60457000
    IF SYMKIND[CURSY]<>TERMINAL THEN                                      60458000
    BEGIN ERROR(13)} INSYMBOL} SKIP(SEMICOLON) END}                       60459000
END OF STATEMENT};                                                        60460000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%70001000
%                                                                         %70002000
%                                                                         %70003000
%                                                                         %70004000
%                  PART 7:   TYPE DECLARATIONS;                           %70005000
%                            ------------------                           %70006000
%                                                                         %70007000
%                                                                         %70008000
%                                                                         %70009000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%70010000
                                                                          70011000
                                                                          70012000
REAL VALX1,VALX2};                                                        70013000
INTEGER TYPEX1,TYPEX2};                                                   70014000
BOOLEAN PACKED};                                                          70015000
                                                                          70016000
PROCEDURE FIELDLIST(RECTAB,FIRSTADDR,LASTADDR)};                          70017000
VALUE RECTAB,FIRSTADDR};                                                  70018000
INTEGER RECTAB,FIRSTADDR,LASTADDR};                                       70019000
FORWARD};                                                                 70020000
                                                                          70021000
DEFINE SUBRANGE=                          %*** SUBRANGE DECLARATION ***   70022000
BEGIN                                     %*****************************   70023000
    CONSTANT(VALX1,TYPEX1)};                                              70024000
    IF TYPETAB[TYPEX1].FORM>CHAR THEN ERROR(11)};                         70025000
```

```
    IF CURSY≠DOUBLEDOT THEN ERROR(53);                                    70026000
    INSYMBOL;                                                             70027000
    CONSTANT(VALX2,TYPEX2);                                               70028000
    IF TYPEX1>0 AND TYPEX2>0 THEN                                         70029000
      IF TYPEX1≠TYPEX2 THEN ERROR(11) ELSE                                70030000
      IF VALX1>VALX2 THEN ERROR(54);                                      70031000
    T1:=TYPETAB1[TYPEX1].FORM; IF T1=SYMBOLIC THEN T1:=SUBTYPE;           70032000
    NEWTYPE; TTYPE:=TYPEINDEX;                                            70033000
    T1.SIZE:=TSIZE:=1; T1.STRUCT:=0; T1.MAINTYPE:=TYPEX1;                 70034000
    TYPETAB1[TYPEINDEX]:=T1;                                              70035000
    TYPETAB2[TYPEINDEX]:=VALX1; TYPETAB3[TYPEINDEX]:=VALX2;               70036000
END OF SUBRANGE#;                                                         70037000
                                                                          70038000
                                                                          70039000
PROCEDURE TYPEDECL(TTYPE,TSIZE);                                          70040000
INTEGER TTYPE,TSIZE;                                                      70041000
BEGIN                                                                     70042000
    PROCEDURE TYPERR(ERRNUM,TTYPE,TSIZE);                                 70043000
    VALUE ERRNUM;                                                         70044000
    INTEGER ERRNUM,TTYPE,TSIZE;                                           70045000
    BEGIN ERROR(ERRNUM);                                                  70046000
          TTYPE:=TSIZE:=0;                                                70047000
    END;                                                                  70048000
                                                                          70049000
    INTEGER RECINX,ARRSTRUCT,TV,SX,T1,T2,T3,T,N;                          70050000
    BOOLEAN FIRST;                                                        70051000
                                                                          70052000
    PACKED:=FALSE;                                                        70080000
    IF CURSY=IDENTIFIER THEN          %*** SIMPLE TYPE DECLARATION ***    70081000
    BEGIN                             %*********************************70082000
      SEARCH;                                                             70083000
      IF FOUND THEN                                                       70084000
      BEGIN                                                               70085000
        IF THISID.IDCLASS=TYPES THEN                                      70086000
        BEGIN                                                             70087000
          TTYPE:=THISID.TYPE; TSIZE:=TYPETAB1[TTYPE].SIZE;                70088000
          INSYMBOL;                                                       70089000
        END ELSE IF THISID.IDCLASS=CONST THEN SUBRANGE                    70090000
                                         ELSE TYPERR(7,TTYPE,TSIZE);      70091000
      END ELSE BEGIN TYPERR(1,TTYPE,TSIZE); INSYMBOL END;                 70092000
    END ELSE                                                              70093000
    IF CURSY≤CHARCONST OR CURSY=PLUS OR CURSY=MINUS THEN SUBRANGE ELSE    70094000
    IF CURSY=LPAR THEN                                                    70095000
    BEGIN                                                                 70096000
      N:=0;                                                               70097000
      NEWTYPE; T3.IDCLASS:=CONST; T3.TYPE:=TYPEINDEX;                     70098000
      DO BEGIN                                                            70099000
        INSYMBOL;                                                         70100000
        IF CURSY=IDENTIFIER THEN                                          70101000
        BEGIN                                                             70102000
          NEWNAME(CURNAME1,CURNAME2,CURLEVEL);                            70103000
          T3.INFO:=N; NAMETAB[CURLEVEL,THISINDEX]:=T3;                    70104000
          N:=N+1; INSYMBOL;                                               70105000
        END ELSE ERROR(9);                                                70106000
      END UNTIL CURSY≠COMMA;                                              70107000
      IF CURSY≠RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;                 70108000
      T1:=SYMBOLIC; T1.STRUCT:=0;                                         70109000
```

```
    T1.SIZE:=TSIZE:=1;  TTYPE:=TYPEINDEX;                                    70110000
    TYPETAB1[TYPEINDEX]:=T1;                                                 70111000
    TYPETAB2[TYPEINDEX]:=0;  TYPETAB3[TYPEINDEX]:=N-1;                       70112000
    IF CURSY=RPAR THEN INSYMBOL;                                             70113000
  END ELSE                                                                   70114000
                                                                             70115000
  IF CURSY=ARROW THEN                    %*** POINTER DECLARATION ***       70116000
  BEGIN                                  %**************************       70117000
    INSYMBOL;                                                                70118000
    IF CURSY=IDENTIFIER THEN                                                 70119000
    BEGIN                                                                    70120000
      NEWTYPE;  TTYPE:=TYPEINDEX;  T1:=POINTERS;                             70121000
      T1.SIZE:=TSIZE:=1;  T1.STRUCT:=0;                                      70122000
      TYPETAB1[TYPEINDEX]:=T1;                                               70123000
      SEARCH;                                                                70124000
      IF FOUND THEN                                                          70125000
      BEGIN                                                                  70126000
        IF THISID.IDCLASS=TYPES THEN                                         70127000
        TYPETAB1[TYPEINDEX].POINTTYPE:=THISID.TYPE ELSE                      70128000
        TYPERR(7,TTYPE,TSIZE);                                               70129000
      END ELSE                                                               70130000
      BEGIN                                                                  70131000
        IF NUMPNTRS<MAXPNTRS THEN NUMPNTRS:=NUMPNTRS+1 ELSE ERROR(52);       70132000
        PNTRTAB1[NUMPNTRS]:=CURNAME1;  PNTRTAB2[NUMPNTRS]:=CURNAME2;         70133000
        PNTRTAB3[NUMPNTRS]:=TYPEINDEX;                                       70134000
      END;                                                                   70135000
      INSYMBOL;                                                              70136000
    END ELSE TYPERR(9,TTYPE,TSIZE);                                          70137000
  END OF POINTER DECLARATION ELSE                                            70138000
  BEGIN                                                                      70139000
    IF CURSY=PACKEDSY THEN BEGIN PACKED:=TRUE; INSYMBOL END;                 70140000
                                                                             70141000
    IF CURSY=ARRAYSY THEN                %*** ARRAY DECLARATION ***          70142000
    BEGIN                                %**************************          70143000
      INSYMBOL;                                                              70144000
      IF CURSY<>LBRACKET THEN ERROR(47) ELSE INSYMBOL;                       70145000
      T:=0;  FIRST:=TRUE;                                                    70146000
      DO BEGIN                                                               70147000
        IF FIRST THEN FIRST:=FALSE ELSE INSYMBOL;                            70148000
        TYPEDECL(TX,SX);                                                     70149000
        IF TX>0 THEN                                                         70150000
        BEGIN                                                                70151000
          IF TYPETAB1[TX].FORM>CHAR THEN ERROR(48);                          70152000
          T1:=ARRAYS;  T1.INXTYPE:=TX;  T1.ARRTYPE:=T;                       70153000
          T2:=TYPETAB2[TX];  T3:=TYPETAB3[TX];                               70154000
          IF T3-T2>1022 THEN ERROR(61);                                      70155000
          T1.SIZE:=MIN(1023,T3-T2+1);                                        70156000
          NEWTYPE;                                                           70157000
          TYPETAB1[TYPEINDEX]:=T1;                                           70158000
          TYPETAB2[TYPEINDEX]:=T2;  TYPETAB3[TYPEINDEX]:=T3;                 70159000
          T:=TYPEINDEX;                                                      70160000
        END;                                                                 70161000
      END UNTIL CURSY<>COMMA;                                                70162000
      IF CURSY<>RBRACKET THEN ERROR(59) ELSE INSYMBOL;                       70163000
      IF CURSY<>OFSY THEN BEGIN ERROR(18); SKIP(OFSY) END;                   70164000
      INSYMBOL;                                                              70165000
      TYPEDECL(TX,SX);                                                       70166000
```

```
  IF TYPETAB1[TX].FORM≥FILES THEN ERROR(60);                          70167000
  ARRSTRUCT:=TYPETAB1[TX].STRUCT;                                     70168000
  WHILE T>0 DO                                                        70169000
  BEGIN                                                               70170000
    T1:=TYPETAB1[T];  T3:=T1.ARRTYPE;                                 70171000
    T1.ARRTYPE:=TX;  T1.STRUCT:=ARRSTRUCT:=ARRSTRUCT+1;               70172000
    T1.SIZE:=SX:=MIN(1023,SX×T1.SIZE);                                70173000
    TYPETAB1[T]:=T1;  TX:=T;  T:=T3;                                  70174000
  END;                                                                70175000
  TTYPE:=TX;  TSIZE:=SX;                                              70176000
END OF ARRAY DECLARATION ELSE                                         70177000
                                                                      70178000
IF CURSY=FILESY THEN                    %*** FILE DECLARATION ***     70179000
BEGIN                                   %************************     70180000
  INSYMBOL;                                                           70181000
  IF CURSY≠OFSY THEN                                                  70182000
  BEGIN ERROR(18);                                                    70183000
        IF CURSY≠IDENTIFIER THEN INSYMBOL;                            70184000
  END ELSE INSYMBOL;                                                  70185000
  TYPEDECL(TX,SX);                                                    70186000
  IF TX>0 THEN                                                        70187000
  BEGIN T:=TYPETAB1[TX];                                              70188000
        IF T.FORM≥FILES THEN ERROR(50) ELSE                           70189000
        IF T.STRUCT>1 THEN ERROR(49);                                 70190000
  END;                                                                70191000
  NEWTYPE;  TTYPE:=TYPEINDEX;                                         70192000
  T1:=IF T.FORM=CHAR THEN TEXTFILE ELSE FILES;                        70193000
  T1.SIZE:=TSIZE:=SX;  T1.FILETYPE:=TX;                               70194000
  T1.STRUCT:=1;                                                       70195000
  TYPETAB1[TYPEINDEX]:=T1;                                            70196000
END OF FILE DECLARATION ELSE                                          70197000
                                                                      70198000
IF CURSY=SETSY THEN                     %*** SET DECLARATION ***      70199000
BEGIN                                   %************************     70200000
  INSYMBOL;                                                           70201000
  IF CURSY≠OFSY THEN                                                  70202000
  BEGIN ERROR(18);                                                    70203000
        IF CURSY>CHARCONST THEN INSYMBOL;                             70204000
  END ELSE INSYMBOL;                                                  70205000
  TYPEDECL(TX,SX);                                                    70206000
  IF TX>0 THEN                                                        70207000
  BEGIN                                                               70208000
    IF TYPETAB1[TX].FORM≠CHAR THEN ERROR(48) ELSE                     70209000
    IF TYPETAB2[TX]<0 OR TYPETAB3[TX]>38 THEN ERROR(51);              70210000
  END;                                                                70211000
  NEWTYPE;  TTYPE:=TYPEINDEX;                                         70212000
  T1:=SET;  T1.SETTYPE:=TX;  T1.STRUCT:=0;                            70213000
  T1.SIZE:=TSIZE:=1;  TYPETAB1[TYPEINDEX]:=T1;                        70214000
  TYPETAB2[TYPEINDEX]:=TYPETAB2[TX];                                  70215000
  TYPETAB3[TYPEINDEX]:=TYPETAB3[TX];                                  70216000
END OF SET DECLARATION ELSE                                           70217000
                                                                      70218000
IF CURSY=RECORDSY THEN                  %*** RECORD DECLARATION ***   70219000
BEGIN                                   %****************************70220000
  IF LASTREC=1>CURLEVEL THEN LASTREC:=LASTREC=1 ELSE ERROR(55);       70221000
  RECINX:=LASTREC;                                                    70222000
  BLOCKTAB[RECINX]:=NUMBLOCKS:=NUMBLOCKS+1;                           70223000
```

```
        INSYMBOL;                                                                70224000
        FIELDLIST(RECINX,0,SX);                                                  70225000
        IF SX>1022 THEN BEGIN ERROR(56); SX:=1022 END;                           70226000
        NEWTYPE; TTYPE:=TYPEINDEX;                                               70227000
        T1:=RECORD; T1.RECTAB:=RECINX; T1.STRUCT:=1;                             70228000
        T1.SIZE:=TSIZE:=SX; TYPETAB1[TYPEINDEX]:=T1;                             70229000
        TYPETAB2[TYPEINDEX]:=0; TYPETAB3[TYPEINDEX]:=SX-1;                       70230000
        IF CURSY#ENDSY THEN BEGIN ERROR(24); SKIP(ENDSY) END;                    70231000
        IF CURSY=ENDSY THEN INSYMBOL;                                            70232000
      END ELSE BEGIN ERROR(4); SKIP(99) END;                                     70233000
    END;                                                                         70234000
END OF TYPEDECL;                                                                 70235000
                                                                                 70236000
                                                                                 70237000
PROCEDURE FIELDLIST(RECTAB,FIRSTADDR,LASTADDR);                                  70238000
VALUE RECTAB,FIRSTADDR;                                                          70239000
INTEGER RECTAB,FIRSTADDR,LASTADDR;                                               70240000
BEGIN                                                                            70241000
    INTEGER ARRAY ILIST[0:LISTLENGTH];                                           70242000
      INTEGER LISTINX;                                                           70243000
    INTEGER CASETYPE,ADDR,MAXADDR,INDEX,CTYPE,TX,SX,T1,T3,LLIM,ULIM,I;           70244000
    BOOLEAN FIRST;                                                               70245000
    REAL CVAL;                                                                   70246000
    LABEL CASETYPEID,CASEPART,EXIT;                                              70247000
                                                                                 70248000
    ADDR:=FIRSTADDR;                                                             70249000
    DO BEGIN                                                                     70250000
      WHILE CURSY=SEMICOLON DO INSYMBOL;                                         70251000
      IF CURSY=CASESY THEN GO TO CASEPART;                                       70252000
      IF CURSY=IDENTIFIER THEN                                                   70253000
      BEGIN                                                                      70254000
        LISTINX:=0; FIRST:=TRUE;                                                 70255000
        DO BEGIN                                                                 70256000
          IF FIRST THEN FIRST:=FALSE ELSE INSYMBOL;                              70257000
          IF CURSY=IDENTIFIER THEN                                               70258000
          BEGIN                                                                  70259000
            IF LISTINX≥LISTLENGTH THEN BEGIN ERROR(37); LISTINX:=0 END;          70260000
            LISTINX:=LISTINX+1;                                                  70261000
            NEWNAME(CURNAME1,CURNAME2,RECTAB);                                   70262000
            ILIST[LISTINX]:=THISINDEX;                                           70263000
            INSYMBOL;                                                            70264000
          END ELSE                                                               70265000
          BEGIN ERROR(9);                                                        70266000
                IF CURSY#COMMA THEN INSYMBOL;                                    70267000
          END;                                                                   70268000
        END UNTIL CURSY#COMMA;                                                   70269000
        IF CURSY#COLON THEN BEGIN ERROR(26); SKIP(COLON) END;                    70270000
        INSYMBOL;                                                                70271000
        TYPEDECL(TX,SX);                                                         70272000
        IF TX>0 THEN IF TYPETAB1[TX].FORM≥FILES THEN ERROR(57);                  70273000
        T3.IDCLASS:=VAR; T3.TYPE:=TX;                                            70274000
        FOR I:=1 STEP 1 UNTIL LISTINX DO                                         70275000
        BEGIN                                                                    70276000
          T3.INFO:=ADDR; ADDR:=MIN(ADDR+SX,1024);                                70277000
          NAMETAB3[RECTAB,ILIST[I]]:=T3;                                         70278000
        END;                                                                     70279000
    END;                                                                         70280000
```

```
    END UNTIL CURSY=SEMICOLON;                                          70281000
    LASTADDR:=ADDR;                                                     70282000
    GO TO EXIT;                                                         70283000
                                                                        70284000
CASEPART:                                                               70285000
    LISTINX:=0;  LASTADDR:=ADDR;  INDEX:=-1;                            70286000
    INSYMBOL;                                                           70287000
    IF CURSY=IDENTIFIER THEN                                            70288000
    BEGIN                                                               70289000
      SEARCH;                                                           70290000
      IF FOUND AND THISID.IDCLASS=TYPES THEN GO TO CASETYPEID;          70291000
      NEWNAME(CURNAME1,CURNAME2,RECTAB);  INDEX:=THISINDEX;             70292000
      INSYMBOL;                                                         70293000
      IF CURSY=COLON THEN ERROR(26);                                    70294000
      INSYMBOL;                                                         70295000
      IF CURSY=IDENTIFIER THEN                                          70296000
      BEGIN                                                             70297000
        SEARCH;                                                         70298000
        IF FOUND THEN                                                   70299000
        BEGIN                                                           70300000
          IF THISID.IDCLASS=TYPES THEN                                  70301000
          BEGIN                                                         70302000
CASETYPEID: CASETYPE:=THISID.TYPE;  T1:=TYPETAB1[CASETYPE];             70303000
            LLIM:=TYPETAB2[CASETYPE];  ULIM:=TYPETAB3[CASETYPE];        70304000
            IF T1.FORM>CHAR THEN ERROR(48);                             70305000
            IF INDEX>=0 THEN                                            70306000
            BEGIN                                                       70307000
              T3.IDCLASS:=VAR;  T3.TYPE:=CASETYPE;  T3.INFO:=ADDR;      70308000
              ADDR:=LASTADDR:=ADDR+1;  NAMETAB3[RECTAB,INDEX]:=T3;      70309000
              END;                                                      70310000
              INSYMBOL;                                                 70311000
          END ELSE BEGIN ERROR(7); SKIP(OFSY) END;                      70312000
        END ELSE BEGIN ERROR(1); SKIP(OFSY) END;                        70313000
      END ELSE BEGIN ERROR(9); SKIP(OFSY) END;                          70314000
    END ELSE BEGIN ERROR(9); SKIP(OFSY) END;                            70315000
    IF CURSY<>OFSY THEN BEGIN ERROR(18); SKIP(RPAR) END;                70316000
    IF CURSY=OFSY THEN INSYMBOL;                                        70317000
    IF CASETYPE=0 THEN BEGIN LLIM:=-MAXINT; ULIM:=MAXINT END;           70318000
    DO BEGIN                                                            70319000
      WHILE CURSY=SEMICOLON DO INSYMBOL;                                70320000
      IF CURSY<=CHARCONST OR CURSY=PLUS OR CURSY=MINUS THEN             70321000
      BEGIN                                                             70322000
        FIRST:=TRUE;                                                    70323000
        DO BEGIN                                                        70324000
          IF FIRST THEN FIRST:=FALSE ELSE INSYMBOL;                     70325000
          CONSTANT(CVAL,CTYPE);                                         70326000
          IF CTYPE>0 THEN                                               70327000
          BEGIN                                                         70328000
            IF CASETYPE=0 THEN CASETYPE:=CTYPE ELSE                     70329000
            IF CVAL<LLIM OR CVAL>ULIM THEN ERROR(14) ELSE               70330000
            CHECKTYPES(CASETYPE,CTYPE);                                 70331000
            IF LISTINX>=LISTLENGTH THEN BEGIN ERROR(30); LISTINX:=0 END; 70332000
            LISTINX:=LISTINX+1;                                         70333000
            ILIST[LISTINX]:=CVAL;  I:=1;                                70334000
            WHILE ILIST[I]<>CVAL DO I:=I+1;                             70335000
            IF I<LISTINX THEN ERROR(31);                                70336000
          END;                                                          70337000
```

```
        END UNTIL CURSY<>COMMA;                                                  70338000
        IF CURSY<>COLON THEN BEGIN ERROR(26); SKIP(LPAR) END;                    70339000
        IF CURSY=COLON THEN INSYMBOL;                                            70340000
        IF CURSY=LPAR THEN                                                       70341000
        BEGIN                                                                    70342000
          INSYMBOL; FIELDLIST(RECTAB,ADDR,MAXADDR);                              70343000
          IF MAXADDR>LASTADDR THEN LASTADDR:=MAXADDR;                            70344000
          IF CURSY<>RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;                   70345000
          INSYMBOL;                                                              70346000
        END ELSE ERROR(58);                                                      70347000
      END;                                                                       70348000
    END UNTIL CURSY = SEMICOLON; %                                               70349000
EXIT:                                                                            70350000
END OF FIELDLIST;                                                                70351000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%80001000
%                                                                               %80002000
%                                                                               %80003000
%                                                                               %80004000
%                  PART 8:   THE PROCEDURE BLOCK.                               %80005000
%                            --------------------                               %80006000
%                                                                               %80007000
%                                                                               %80008000
%                                                                               %80009000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%80010000
                                                                                 80011000
                                                                                 80012000
                                                                                 80013000
PROCEDURE DECLAREVARS(PARAM,TAB,FIRST,LAST,LEVEL);                               80014000
VALUE PARAM,FIRST,LAST,LEVEL;                                                    80015000
INTEGER ARRAY TAB[0];                                                            80016000
INTEGER FIRST,LAST,LEVEL;                                                        80017000
BOOLEAN PARAM;                                                                   80018000
BEGIN                                                                            80019000
    INTEGER LEVEL1000,TYP,NAM,NAMTAB,T1,I,J,RECSIZE;                             80020000
    BOOLEAN REALVAR,ARRAYVAR,FIRSTDIM,EXTFILE;                                   80021000
    ALPHA FNAME;                                                                 80022000
    INTEGER FNLENGTH,FNSTART; %                                                  80023000
                                                                                 80024000
    LEVEL1000:=LEVEL*1000;                                                       80025000
    FOR I:=FIRST STEP 1 UNTIL LAST DO                                            80026000
    BEGIN                                                                        80027000
      NAM:=TAB[I],[9:10]; NAMTAB:=NAMETAB3[LEVEL,NAM];                           80028000
      TYP:=NAMTAB.TYPE;      T1:=TYPETAB1[TYP];                                  80029000
      IF NAMTAB.IDCLASS GEQ FUNC THEN                                            80030000
      BEGIN                                                                      80031000
        IF REALVAR OR ARRAYVAR THEN                                              80032000
        BEGIN                                                                    80033000
          GEN("]",1,7);                                                          80034000
          REALVAR:=ARRAYVAR:=FALSE;                                              80035000
        END;                                                                     80036000
        IF NAMTAB.IDCLASS=FUNC THEN GEN("REAL",5,4);                             80037000
        GEN("PROCEDU",8,1);                                                      80038000
        GENIDC("V",LEVEL1000+NAM,5); GEN("]",1,7);                               80039000
      END ELSE                                                                   80040000
      IF T1.STRUCT=0 THEN                  %*** SIMPLE TYPE ***                80041000
      BEGIN                                                                      80042000
        IF ARRAYVAR THEN BEGIN GEN("]",1,7); ARRAYVAR:=FALSE END;                80043000
```

```
  IF REALVAR THEN GEN(";",1,7) ELSE                                    80044000
      BEGIN GEN("REAL",5,4); REALVAR:=TRUE END;                        80045000
  GENID("V",LEVEL1000+NAM,5);                                          80046000
END ELSE                                                               80047000
BEGIN                                                                  80048000
  IF REALVAR THEN BEGIN GEN("]",1,7); REALVAR:=FALSE END;              80049000
  IF T1.FORM<FILES THEN            %*** ARRAY/RECORD ***               80050000
  BEGIN                                                                80051000
    IF ARRAYVAR THEN GEN(",",1,7) ELSE                                 80052000
        BEGIN GEN("ARRAY",6,3); ARRAYVAR:=TRUE END;                    80053000
    GENID("V",LEVEL1000+NAM,5);  GEN("[",1,7);                         80054000
    FIRSTDIM:=TRUE;                                                    80055000
    DO BEGIN                                                           80056000
      IF FIRSTDIM THEN FIRSTDIM:=FALSE ELSE GEN(",",1,7);              80057000
      GENINT(TYPETAB2[TYP]);                                           80058000
      IF NOT PARAM THEN                                                80059000
      BEGIN GEN(":",1,7); GENINT(TYPETAB3[TYP]) END;                   80060000
      TYP:=IF T1.FORM=ARRAYS THEN T1.ARRTYPE ELSE REALTYPE;            80061000
      T1:=TYPETAB1[TYP];                                               80062000
    END UNTIL T1.STRUCT=0;                                             80063000
    GEN("]",1,7);                                                      80064000
  END ELSE                                                             80065000
  BEGIN                            %*** FILE ***                       80066000
    IF REALVAR OR ARRAYVAR THEN                                        80067000
        BEGIN GEN("]",1,7); REALVAR:=ARRAYVAR:=FALSE END;              80068000
    IF T1.FORM=TEXTFILE AND NOT PARAM THEN                             80069000
BEGIN                                                                  80070000
  IF NUMFILES>MAXFILES THEN ERROR(97)                                  80071000
                       ELSE NUMFILES:=NUMFILES+1;                      80072000
  FILETAB[NUMFILES]:=NAM;                                              80073000
END;                                                                   80074000
EXTFILE:=FALSE;                                                        80075000
FNAME:=NAMETAB1[LEVEL,NAM];                                            80076000
FNLENGTH := FNAME.NAMELENGTH;  FNSTART := 8-FNLENGTH; %                80077000
IF FNLENGTH LEQ 6 THEN %                                               80078000
BEGIN                                                                  80079000
  FOR J:=1 STEP 1 UNTIL NUMEXTFILES DO                                 80080000
  IF FNAME=EXTFILETAB[J] THEN EXTFILE:=TRUE;                           80081000
END;                                                                   80082000
IF EXTFILE AND NOT PARAM THEN                                          80083000
BEGIN                                                                  80084000
   IF NUMFILES GEQ MAXFILES THEN ERROR(97)                             80085000
   ELSE                                                                80086000
   NUMFILES := NUMFILES + 1;                                           80087000
   FILETAB[NUMFILES] := -NAM - 1;                                      80088000
  GEN("DEFINE",7,2);  GENID("F",LEVEL1000+NAM,5);                      80089000
  GEN("=",1,7);                                                        80090000
  GEN(FNAME,FNLENGTH,FNSTART); %                                       80091000
  GEN("#;",2,6);   GEN("SAVE",5,4);    GEN("FILE",5,4);                80092000
  GEN(FNAME,FNLENGTH,FNSTART); %                                       80093000
END ELSE                                                               80094000
BEGIN                                                                  80095000
  GEN("FILE",5,4);  GENID("F",LEVEL1000+NAM,5);                        80096000
END;                                                                   80097000
IF NOT PARAM THEN                                                      80098000
BEGIN                                                                  80099000
  GEN(" DISK",6,3);   GEN("SERIAL",7,2);                               80100000
```

```
    IF FXTFILE THEN                                                          80101000
    BEGIN                                                                    80102000
      IF ALGOLCNT LSS 13 THEN WRITEALGOL;                                    80103000
       GEN("[0:0]",5,3);                                                     80104000
      GEN("""",1,7);                                                         80105000
      GEN(FNAME,FNLENGTH,FNSTART); %                                         80106000
      GEN("""",1,7);     GEN("/",1,7);                                       80107000
      IF ALGOLCNT<9 THEN WRITEALGOL;                                         80108000
      GEN("""",1,7);     GEN(USER,7,1);     GEN("""",1,7);                  80109000
    END ELSE                                                                 80110000
    BEGIN                                                                    80111000
      GEN("[20:",4,4);  GEN("300]",4,4);                                     80112000
    END;                                                                     80113000
    GEN("(1,",3,5);                                                          80114000
    RECSIZE:=IF T1.FORM=TEXTFILE THEN 10 ELSE                                80115000
             IF TYPETAB1[T1.FILETYPE].STRUCT=0 THEN 1 ELSE                   80116000
               TYPETAB2[T1.FILETYPE]=TYPETAB2[T1.FILETYPE]+1;                80117000
    GENINT(RECSIZE);    GEN(",",1,7);                                        80118000
    IF RECSIZE=1 OR RECSIZE=10 THEN GENINT(150)                              80119000
                               ELSE GENINT(RECSIZE);                         80120000
    IF ALGOLCNT LSS 10 THEN WRITEALGOL;                                      80121000
    GEN(",SAVE",6,3);   GEN("30",2,6);                                       80122000
    GEN(")]",2,6);                                                           80123000
END ELSE GEN("]",1,7);                                                       80124000
GEN("ARRAY",6,3);     GENID("V",LEVEL1000+NAM,5);                            80125000
GEN("[",1,7);                                                                80126000
IF TYPETAB1[T1.FILETYPE].STRUCT=0 THEN                                       80127000
BEGIN                                                                        80128000
  IF PARAM THEN GEN("0",1,7) ELSE GEN("0:0",3,5);                            80129000
END ELSE                                                                     80130000
          BEGIN                                                              80131000
            GENINT(TYPETAB2[T1.FILETYPE]);                                   80132000
            IF NOT PARAM THEN                                                80133000
            BEGIN GEN(":",1,7); GENINT(TYPETAB3[T1.FILETYPE]) END;           80134000
          END;                                                               80135000
          GEN("]",2,6);                                                      80136000
          GEN("INTEGER",8,1);   GENID("I",LEVEL1000+NAM,5);                  80137000
          GEN(";",1,7);                                                      80138000
        END;                                                                 80139000
      END;                                                                   80140000
    END OF LOOP;                                                             80141000
    IF REALVAR OR ARRAYVAR THEN GEN("]",1,7);                                80142000
END OF DECLAREVARS;                                                          80143000
                                                                             80144000
                                                                             80145000
PROCEDURE PARAMETERLIST;                                                     80146000
BEGIN                                                                        80147000
    INTEGER FIRSTPARAM,CURKIND,P1,PX,I,T;                                    80148000
    BOOLEAN FIRST;                                                           80149000
                                                                             80150000
    DEFINE NEWPARAM=                                                         80151000
    BEGIN                                                                    80152000
        IF NUMPARAMS≥MAXPARAMS THEN                                          80153000
        BEGIN ERROR(70); NUMPARAMS:=MAXPARAMS-10 END;                        80154000
        NUMPARAMS:=NUMPARAMS+1;                                              80155000
    END OF NEWPARAM#;                                                        80156000
                                                                             80157000
```

```
NEWPARAM; FIRSTPARAM:=NUMPARAMS;                                        80158000
IF CURSY=LPAR THEN                                                      80159000
BEGIN                                                                   80160000
  DO BEGIN                                                              80161000
    INSYMBOL;                                                           80162000
    IF CURSY=VARSY OR CURSY=FUNCSY OR CURSY=PROCSY THEN                 80163000
    BEGIN                                                               80164000
      CURKIND:=IF CURSY=VARSY  THEN VAR  ELSE                           80165000
               IF CURSY=FUNCSY THEN FUNC ELSE PROC;                     80166000
      INSYMBOL;                                                         80167000
    END ELSE CURKIND:=CONST;                                            80168000
    FIRST:=TRUE; P1:=NUMPARAMS+1;                                       80169000
    DO BEGIN                                                            80170000
      IF FIRST THEN FIRST:=FALSE ELSE INSYMBOL;                         80171000
      IF CURSY=IDENTIFIER THEN                                          80172000
      BEGIN                                                             80173000
        NEWNAME(CURNAME1,CURNAME2,CURLEVEL+1);                          80174000
        PX:=THISINDEX; PX.PARAMKIND:=CURKIND;                           80175000
        PX.PARAMLEVEL:=CURLEVEL+1;                                      80176000
        NEWPARAM; PARAMTAB[NUMPARAMS]:=PX;                              80177000
      END ELSE ERROR(9);                                                80178000
      INSYMBOL;                                                         80179000
    END UNTIL CURSY#COMMA;                                              80180000
    IF CURSY=COLON THEN                                                 80181000
    BEGIN                                                               80182000
      IF CURKIND=PROC THEN ERROR(90);                                   80183000
      INSYMBOL;                                                         80184000
      IF CURSY=IDENTIFIER THEN                                          80185000
      BEGIN                                                             80186000
        SEARCH;                                                         80187000
        IF FOUND THEN                                                   80188000
        BEGIN                                                           80189000
          IF THISID.IDCLASS=TYPES THEN                                  80190000
          BEGIN                                                         80191000
            T3:=THISID.TYPE;                                            80192000
            FOR I:=P1 STEP 1 UNTIL NUMPARAMS DO                         80193000
            PARAMTAB[I].PARAMTYPE:=T3;                                  80194000
            IF CURKIND=CONST OR CURKIND=VAR THEN                        80195000
            BEGIN                                                       80196000
              T:=TYPETAB1[T3];                                          80197000
              IF T.FORM>=FILES THEN                                     80198000
                FOR I:=P1 STEP 1 UNTIL NUMPARAMS DO                     80199000
                PARAMTAB[I].PARAMFILE:=1;                               80200000
              IF T.STRUCT>0 AND CURKIND=CONST THEN ERROR(94);           80201000
            END ELSE IF T.STRUCT>0 THEN ERROR(38);                      80202000
          END ELSE BEGIN ERROR(7); T3:=0 END;                           80203000
        END ELSE BEGIN ERROR(1); T3:=0 END;                             80204000
      END ELSE BEGIN ERROR(9); T3:=0 END;                               80205000
      INSYMBOL;                                                         80206000
    END ELSE                                                            80207000
    BEGIN                                                               80208000
      IF CURKIND#PROC THEN ERROR(7);                                    80209000
      T3:=0;                                                            80210000
    END;                                                                80211000
    T3.IDCLASS:=CURKIND; T3.FORMAL:=1;                                  80212000
    FOR I:=P1 STEP 1 UNTIL NUMPARAMS DO                                 80213000
    NAMETAB[CURLEVEL+1,PARAMTAB[I].PARAMNAME]:=T3;                      80214000
```

```
      END UNTIL CURSY#SEMICOLON;                                                    80215000
      IF CURSY#RPAR THEN                                                            80216000
      BEGIN ERROR(46);  SKIP(RPAR);                                                 80217000
            IF CURSY=RPAR THEN INSYMBOL;                                            80218000
      END ELSE INSYMBOL;                                                            80219000
    END;                                                                            80220000
    PARAMTAB[FIRSTPARAM]:=NUMPARAMS-FIRSTPARAM;                                     80221000
END OF PARAMETERLIST;                                                               80222000
                                                                                    80223000
                                                                                    80400000
PROCEDURE BLOCK;                                                                    80401000
BEGIN                                                                               80402000
    INTEGER INDEX,CTYPE,NUMFORWARDS,T,T3,TX,I;                                      80403000
    REAL CVAL;                                                                      80404000
    ALPHA C1,C2;                                                                    80405000
    BOOLEAN VALUEPARAMS,FUN;                                                        80406000
    LABEL START;                                                                    80407000
                                                                                    80408000
    INTEGER LABTABTOP,CONSTTABTOP,TYPETABTOP,PARAMTABTOP,TOPREC,                    80409000
    FORMERFIRSTLAB,FIRSTFILE;                                                       80410000
                                                                                    80411000
    FORMERFIRSTLAB:=FIRSTLAB;                                                       80412000
    LABTABTOP:=NUMLABS;   FIRSTLAB:=LABTABTOP+1;                                    80413000
    CONSTTABTOP:=NUMCONSTS;                                                         80414000
    TYPETABTOP:=NUMTYPES;                                                           80415000
    PARAMTABTOP:=NUMPARAMS;                                                         80416000
    TOPREC:=LASTREC;                                                                80417000
    FIRSTFILE:=NUMFILES+1;                                                          80418000
                                                                                    80419000
    TOPLEVEL:=CURLEVEL;                                                             80420000
    IF CURLEVEL>1 THEN GEN("BEGIN",6,3);                                            80421000
START;                                                                              80422000
    IF CURSY=LABELSY THEN                  %*** LABEL DECLARATION ***               80423000
    BEGIN                                  %*************************               80424000
        GEN("LABEL",6,3);                                                           80425000
        DO BEGIN                                                                    80426000
          INSYMBOL;                                                                 80427000
          IF CURSY=INTCONST THEN                                                    80428000
          BEGIN                                                                     80429000
            GENID("L",CURVAL,4);                                                    80430000
            IF CURVAL>9999 THEN ERROR(33);                                          80431000
            FOR I:=FIRSTLAB STEP 1 UNTIL NUMLABS DO                                 80432000
            IF LABTAB[I].LABVAL=CURVAL THEN ERROR(31);                              80433000
            IF NUMLABS≥MAXLABS THEN BEGIN ERROR(34); NUMLABS:=0 END;                80434000
            NUMLABS:=NUMLABS+1;                                                     80435000
            LABTAB[NUMLABS]:=CURVAL;                                                80436000
            INSYMBOL;                                                               80437000
          END ELSE BEGIN ERROR(10); SKIP(COMMA) END;                                80438000
          IF CURSY=COMMA THEN GEN(",",1,7);                                         80439000
        END UNTIL CURSY#COMMA;                                                      80440000
        IF CURSY#SEMICOLON THEN BEGIN ERROR(25); SKIP(SEMICOLON) END;               80441000
        GEN(";",1,7);                                                               80442000
        IF SYMKIND[CURSY]#INITIAL THEN INSYMBOL;                                    80443000
    END OF LABEL DECLARATION;                                                       80444000
                                                                                    80445000
    IF CURSY=CONSTSY THEN                  %*** CONSTANT DECLARATION ***            80446000
    BEGIN                                  %*****************************           80447000
```

```
      INSYMBOL;                                                                  80448000
      DO BEGIN                                                                   80449000
        IF CURSY=IDENTIFIER THEN                                                 80450000
        BEGIN                                                                    80451000
          NEWNAME(CURNAME1,CURNAME2,CURLEVEL);  INDEX:=THISINDEX;                80452000
          INSYMBOL;                                                              80453000
          IF CURSY=EQLSY THEN                                                    80454000
          BEGIN                                                                  80455000
            INSYMBOL;  CONSTANT(CVAL,CTYPE);                                     80456000
            T3:=CTYPE;  T3.IDCLASS:=CONST;                                       80457000
            IF CVAL.[46:81]#0 OR CVAL>1023 THEN                                  80458000
            BEGIN                                                                80459000
              IF NUMCONSTS≥MAXCONSTS THEN                                        80460000
              BEGIN ERROR(35); NUMCONSTS:=0 END;                                 80461000
              NUMCONSTS:=NUMCONSTS+1;                                            80462000
              CONSTTAB[NUMCONSTS]:=CVAL;                                         80463000
              T3.INFO:=1023+NUMCONSTS;                                           80464000
            END ELSE T3.INFO:=CVAL;                                              80465000
            NAMETAB[CURLEVEL,INDEX]:=T3;                                         80466000
          END ELSE BEGIN ERROR(36); SKIP(SEMICOLON) END;                         80467000
        END ELSE BEGIN ERROR(9); SKIP(SEMICOLON) END;                            80468000
        IF CURSY#SEMICOLON THEN BEGIN ERROR(25); SKIP(SEMICOLON) END;            80469000
      IF SYMKIND[CURSY]#INITIAL THEN INSYMBOL;                                   80470000
      END UNTIL CURSY#IDENTIFIER;                                                80471000
  END OF CONSTANT DECLARATION;                                                   80472000
                                                                                 80473000
  IF CURSY=TYPESY THEN                    %*** TYPE DECLARATION ***              80474000
  BEGIN                                   %************************              80475000
      INSYMBOL;                                                                  80476000
      DO BEGIN                                                                   80477000
        IF CURSY=IDENTIFIER THEN                                                 80478000
        BEGIN                                                                    80479000
          NEWNAME(CURNAME1,CURNAME2,CURLEVEL);  INDEX:=THISINDEX;                80480000
          INSYMBOL;                                                              80481000
          IF CURSY=EQLSY THEN                                                    80482000
          BEGIN                                                                  80483000
            INSYMBOL;                                                            80484000
            TYPEDECL(CTYPE,TY);                                                  80485000
            T3:=CTYPE; T3.IDCLASS:=TYPES;                                        80486000
            NAMETAB[CURLEVEL,INDEX]:=T3;                                         80487000
          END ELSE BEGIN ERROR(36); SKIP(SEMICOLON) END;                         80488000
        END ELSE BEGIN ERROR(9); SKIP(SEMICOLON) END;                            80489000
        IF CURSY#SEMICOLON THEN BEGIN ERROR(25); SKIP(SEMICOLON) END;            80490000
      IF SYMKIND[CURSY]#INITIAL THEN INSYMBOL;                                   80491000
      END UNTIL CURSY#IDENTIFIER;                                                80492000
  END OF TYPE DECLARATION;                                                       80493000
                                                                                 80494000
  IF CURSY=VARSY THEN                     %*** VARIABLE DECLARATION ***          80495000
  BEGIN                                   %****************************          80496000
      VARINDEX:=0;                                                               80497000
      DO BEGIN                                                                   80498000
        FIRSTVAR:=VARINDEX+1;                                                    80499000
        DO BEGIN                                                                 80500000
          IF CURSY=VARSY OR CURSY=COMMA THEN INSYMBOL;                           80501000
          IF CURSY=IDENTIFIER THEN                                               80502000
          BEGIN                                                                  80503000
            IF VARINDEX≥LISTLENGTH THEN                                          80504000
```

```
            BEGIN ERROR(37); VARINDEX:=0 END;                                  80505000
            VARINDEX:=VARINDEX+1;                                              80506000
            NEWNAME(CURNAME1.CURNAME2,CURLEVEL);                               80507000
            VARLIST[VARINDEX]:=THISINDEX;                                      80508000
            INSYMBOL;                                                          80509000
          END ELSE BEGIN ERROR(9); SKIP(COLON) END;                            80510000
        END UNTIL CURSY<>COMMA;                                                80511000
        IF CURSY<>COLON THEN BEGIN ERROR(26); SKIP(COLON) END;                 80512000
        IF CURSY=COLON THEN                                                    80513000
        BEGIN                                                                  80514000
          INSYMBOL;                                                            80515000
          TYPEDECL(CTYPE,TX);                                                  80516000
          T3:=CTYPE;  T3.IDCLASS:=VAR;                                         80517000
          FOR I:=FIRSTVAR STEP 1 UNTIL VARINDEX DO                             80518000
            NAMETAB3[CURLEVEL,VARLIST[I]]:=T3;                                 80519000
        END ELSE BEGIN ERROR(26); SKIP(SEMICOLON) END;                         80520000
        IF CURSY<>SEMICOLON THEN BEGIN ERROR(25); SKIP(SEMICOLON) END;         80521000
      IF SYMKIND[CURSY]<>INITIAL THEN INSYMBOL;                                80522000
      END UNTIL CURSY<>IDENTIFIER;                                             80523000
      DECLAREVARS(FALSE,VARLIST,1,VARINDEX,CURLEVEL);                          80524000
END OF VARIABLE DECLARATION;                                                   80525000
                                                                               80526000
IF NUMPNTRS>0 THEN                                                             80527000
BEGIN                                                                          80528000
  C1:=CURNAME1;  C2:=CURNAME2;                                                 80529000
  FOR I:=1 STEP 1 UNTIL NUMPNTRS DO                                            80530000
  BEGIN                                                                        80531000
    CURNAME1:=PNTRTAB1[I];  CURNAME2:=PNTRTAB2[I];                             80532000
    SEARCHTAB(CURLEVEL);                                                       80533000
    THISID:=NAMETAB3[CURLEVEL,THISINDEX];                                      80534000
    IF FOUND AND THISID.IDCLASS=TYPES THEN                                     80535000
    TYPETAB[PNTRTAB3[I]].POINTTYPE:=THISID.TYPE ELSE ERROR(62);                80536000
  END;                                                                         80537000
  CURNAME1:=C1;  CURNAME2:=C2;  NUMPNTRS:=0;                                   80538000
END;                                                                           80539000
                                                                               80540000
WHILE CURSY=FUNCSY OR CURSY=PROCSY DO %*** PROC/FUNC DECLARATION ***           80541000
BEGIN                                 %*********************************       80542000
    FUN:=CURSY=FUNCSY;  INSYMBOL;                                              80543000
    IF CURSY=IDENTIFIER THEN                                                   80544000
    BEGIN                                                                      80545000
      SEARCHTAB(CURLEVEL);                                                     80546000
      THISID:=NAMETAB3[CURLEVEL,THISINDEX];                                    80547000
      IF FOUND AND THISID.IDCLASS>=PROC THEN                                   80548000
      BEGIN                                                                    80549000
        INDEX:=THISINDEX;                                                      80550000
        IF THISID.FORWARDDEF=1 THEN                                            80551000
        BEGIN                                                                  80552000
          NAMETAB3[THISLEVEL,THISINDEX].FORWARDDEF:=0;                         80553000
          NUMFORWARDS:=NUMFORWARDS-1;                                          80554000
          IF(THISID.IDCLASS<>PROC AND FUN)OR                                   80555000
            (THISID.IDCLASS<>FUNC AND NOT FUN) THEN ERROR(43);                 80556000
          INSYMBOL;                                                            80567000
        END ELSE BEGIN ERROR(2); SKIP(SEMICOLON) END;                          80568000
      END ELSE                                                                 80569000
      BEGIN                                                                    80570000
        NEWNAME(CURNAME1,CURNAME2,CURLEVEL);  INDEX:=THISINDEX;                80571000
```

```
T3:=0;  T3.INFO:=NUMPARAMS+1;                                          80572000
T3.IDCLASS:=IF FUN THEN FUNC ELSE PROC;                                80573000
NAMETAB[CURLEVEL,INDEX]:=T3;                                           80574000
INSYMBOL;  PARAMETERLIST;                                              80575000
IF CURSY=COLON THEN                                                    80576000
BEGIN                                                                  80577000
  IF NOT FUN THEN ERROR(48);                                           80578000
  INSYMBOL;                                                            80579000
  IF CURSY=IDENTIFIER THEN                                             80580000
  BEGIN                                                                80581000
    SEARCH;                                                            80582000
    IF FOUND THEN                                                      80583000
    BEGIN                                                              80584000
      IF THISID.IDCLASS=TYPES THEN                                     80585000
      BEGIN                                                            80586000
        T:=TYPETAB[THISID.TYPE];                                       80587000
        IF T.FORM=ALFA OR T.FORM=POINTERS THEN                         80588000
        BEGIN                                                          80589000
          NAMETAB[CURLEVEL,INDEX].TYPE:=THISID.TYPE;                   80590000
        END ELSE ERROR(38);                                            80591000
      END ELSE ERROR(7);                                               80592000
    END ELSE ERROR(1);                                                 80593000
  END ELSE ERROR(9);                                                   80594000
  INSYMBOL;                                                            80595000
END ELSE IF FUN THEN                                                   80596000
BEGIN ERROR(26);  SKIP(SEMICOLON) END;                                 80597000
  END;                                                                 80598000
END ELSE BEGIN ERROR(9); SKIP(SEMICOLON) END;                          80599000
IF CURSY<>SEMICOLON THEN BEGIN ERROR(25); SKIP(SEMICOLON) END;         80600000
IF FUN THEN GEN("FUNCTN",7,2) ELSE                                     80601000
GEN("PROCEDU",8,1);  GENID("V",1000*CURLEVEL+INDEX,5);                 80602000
T:=NAMETAB[CURLEVEL,INDEX].INFO;  TX:=T+PARAMTAB[T];                   80603000
IF TX>T THEN                                                           80604000
BEGIN                                                                  80605000
  GEN("(",1,7);                                                        80606000
  FOR I:=T+1 STEP 1 UNTIL TX DO                                        80607000
  BEGIN GENID("V",1000*(CURLEVEL+1)+PARAMTAB[I].PARAMNAME,5);          80608000
    IF BOOLEAN(PARAMTAB[I].PARAMFILE) THEN                             80609000
    BEGIN                                                              80610000
      GEN(",",1,7);                                                    80611000
      GENID("F",1000*(CURLEVEL+1)+PARAMTAB[I].PARAMNAME,5);            80612000
      GEN(",",1,7);                                                    80613000
      GENID("I",1000*(CURLEVEL+1)+PARAMTAB[I].PARAMNAME,5);            80614000
    END;                                                               80615000
    IF I LSS TX THEN GEN(",",1,7);                                     80616000
  END;                                                                 80617000
  GEN(")",2,6);                                                        80618000
  VALUEPARAMS:=FALSE;                                                  80619000
  FOR I:=T+1 STEP 1 UNTIL TX DO                                        80620000
  IF PARAMTAB[I].PARAMKIND=CONST THEN                                  80621000
  BEGIN                                                                80622000
    IF NOT VALUEPARAMS THEN                                            80623000
    BEGIN GEN("VALUE",4,3);                                            80624000
          VALUEPARAMS:=TRUE;                                           80625000
    END ELSE GEN(",",1,7);                                             80626000
    GENID("V",1000*(CURLEVEL+1)+PARAMTAB[I].PARAMNAME,5);              80627000
  END;                                                                 80628000
```

```
      IF VALUEPARAMS THEN GEN("}",1,7);                                    80629000
      DECLAREVARS(TRUE,PARAMTAB,T+1,TX,CURLEVEL+1);                        80630000
    END ELSE GEN("}",1,7);                                                 80631000
                                                                           80632000
    INSYMBOL;                                                              80633000
    IF CURNAME1="FORWAR" AND CURNAME2="D" THEN                             80634000
    BEGIN                                                                  80635000
      NAMETAB3[CURLEVEL,INDEX].FORWARDDEF:=1;                              80636000
      NUMFORWARDS:=NUMFORWARDS+1;                                          80637000
      GEN("FORWARD",8,1);                                                  80638000
      INSYMBOL;                                                            80639000
    END ELSE                                                               80640000
    BEGIN                                                                  80641000
      CURLEVEL:=CURLEVEL+1;                                                80642000
      IF CURLEVEL≥LASTREC THEN ERROR(55);                                  80643000
      BLOCKTAB[CURLEVEL]:=NUMBLOCKS:=NUMBLOCKS+1;                          80644000
      T:=CURFUNC; CURFUNC:=IF FUN THEN INDEX ELSE -1;                      80645000
      BLOCK;   %*** COMPILE PROCEDURE BODY ***                             80646000
      REPLACE POINTER(NAMETAB[CURLEVEL,*]) BY 0                            80647000
                                    FOR MAXNAMES+1 WORDS;                  80648000
      CURLEVEL:=CURLEVEL-1; CURFUNC:=T;                                    80649000
      TOPLEVEL:=CURLEVEL;                                                  80650000
    END;                                                                   80651000
    IF CURSY≠SEMICOLON THEN BEGIN ERROR(25); SKIP(SEMICOLON) END;          80652000
    GEN(";",1,7);                                                          80653000
    IF SYMKIND[CURSY]≠INITIAL THEN INSYMBOL;                               80654000
END OF PROCEDURE DECLARATION;                                              80655000
                                                                           80656000
                                                                           80657000
IF NUMFORWARDS>0 THEN ERROR(44);                                           80658000
GEN("INTEGER",8,1);                                                        80659000
FOR I:=1 STEP 1 UNTIL MAXTEMPS DO                                          80660000
BEGIN GENID("T",I,2);                                                      80661000
      IF I<MAXTEMPS THEN GEN(",",1,7) ELSE GEN(";",1,7);                   80662000
END;                                                                       80663000
IF CURSY≠BEGINSY THEN                                                      80664000
BEGIN ERROR(39);                                                           80665000
      WHILE SYMKIND[CURSY]≠INITIAL DO                                      80666000
      BEGIN INSYMBOL; SKIP(SEMICOLON) END;                                 80667000
      IF(CURSY=TYPESY)OR(CONSTSY≤CURSY AND CURSY≤PROCSY)THEN               80668000
      GO TO START;                                                         80669000
END;                                                                       80670000
IF CURLEVEL=1 THEN                                                         80671000
BEGIN                                                                      80672000
  GEN("INIT(",5,3);                                                        80673000
  IF INPUTDECL THEN GEN("TRUE",4,4) ELSE GEN("FALSE",5,3);                 80674000
  GEN(");",2,6);                                                           80675000
END;                                                                       80676000
FOR I:=FIRSTFILE STEP 1 UNTIL NUMFILES DO                                  80677000
 IF FILETAB[I] LSS 0 THEN                                                  80678000
 BEGIN                                                                     80679000
     GEN("CWFIL(",6,2);                                                    80680000
     GENID("F",1000×CURLEVEL-FILETAB[I]-1,5);                              80681000
     GEN(");",2,6);                                                        80682000
 END                                                                       80683000
 ELSE                                                                      80684000
BEGIN                                                                      80685000
```

```
      GENID("I",1000*CURLEVEL+FILETAB[I],5);                                  80686000
      GEN(",",1,7);  GEN("BUFSIZE",7,1);  GEN(":=80;",5,3);                  80687000
    END;                                                                      80688000
    NUMFILES:=FIRSTFILE-1;                                                    80689000
                                                                              80690000
    COMPSTAT;      **** COMPILE STATEMENT PART ***                            80691000
                                                                              80692000
    FOR I:=LASTREC STEP 1 UNTIL TOPREC-1 DO                                   80693000
    REPLACE POINTER(NAMETAB[I,*]) BY 0 FOR MAXNAMES+1 WORDS;                  80694000
    FOR I:=FIRSTLAB STEP 1 UNTIL NUMLABS DO                                   80695000
    IF LABTAB[I].LABDEF=0 THEN ERROR(93);                                     80696000
    LASTREC:=TOPREC;                                                          80697000
    NUMLABS:=LABTABTOP;                                                       80698000
    FIRSTLAB:=FORMERFIRSTLAB;                                                 80699000
    NUMCONSTS:=CONSTTABTOP;                                                   80700000
    NUMTYPES:=TYPETABTOP;                                                     80701000
    NUMPARAMS:=PARAMTABTOP;                                                   80702000
    IF CURLEVEL>1 THEN GEN("END",4,5);                                        80703000
END OF BLOCK;                                                                 80704000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%90001000
%                                                                             %90002000
%                                                                             %90003000
%                                                                             %90004000
%                  PART 9:   THE MAIN PROGRAM;                                %90005000
%                            ---------------                                  %90006000
%                                                                             %90007000
%                                                                             %90008000
%                                                                             %90009000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%90010000
                                                                              90011000
                                                                              90012000
INTEGER PROGNAMELENGTH;                                                       90013000
ALPHA PROGNAME,ALGOLNAME;                                                     90014000
                                                                              90015000
ALGOLNAME:="PASC000"&ENTIER(TIME(4)MOD 10)[17:5:6];                           90016000
ALGOLNAME:=ALGOLNAME&ENTIER(TIME(4)DIV 7)[11:5:6];                            90017000
ALGOLNAME:=ALGOLNAME&ENTIER(TIME(4)MOD 9)[5:5:6];                             90018000
USER:=TIME(-1);                                                               90019000
FILL PASCALGOL WITH ALGOLNAME,USER;                                           90020000
BEGIN                                                                         90021000
    FILE PASCRUN DISK SERIAL "PASCRUN"/"DISK" (2,10,150);                     90022000
    ARRAY BUF[0:9];                                                           90023000
    LABEL EOF;                                                                90024000
                                                                              90025000
    WHILE TRUE DO                                                             90026000
    BEGIN                                                                     90027000
      READ(PASCRUN,9,BUF[*]) [EOF];                                           90028000
      WRITE(PASCALGOL,10,BUF[*]);                                             90029000
    END;                                                                      90030000
EOF:                                                                          90031000
END OF TRANSFER OF RUN TIME SYSTEM;                                           90032000
CARDLENGTH:=72;                                                               90033000
INITIALIZE;  NEWCARD;                                                         90034000
LISTOPTION:=CHECKOPTION:=TRUE;                                                90035000
C:=" ";  INSYMBOL;                                                           90036000
IF CURSY=PROGRAMSY THEN                                                       90037000
BEGIN                                                                         90038000
```

```
    INSYMBOL;                                                                   90039000
    IF CURSY=IDENTIFIER THEN                                                    90040000
    BEGIN                                                                       90041000
      PROGNAME:=CURNAME1.[35:36];  PROGNAMELENGTH:=MIN(6,CURLENGTH);            90042000
      INSYMBOL;                                                                 90043000
      IF CURSY=LPAR THEN                                                        90044000
      BEGIN                                                                     90045000
        DO BEGIN                                                                90046000
          INSYMBOL;                                                             90047000
          IF CURSY=IDENTIFIER THEN                                              90048000
          BEGIN                                                                 90049000
            IF CURNAME1="5INPUT" THEN INPUTDECL:=TRUE ELSE                      90050000
            IF CURNAME1="6OUTPUT" THEN OUTPUTDECL:=TRUE ELSE                    90051000
            BEGIN                                                               90052000
              IF CURLENGTH>6 THEN ERROR(77);                                    90053000
              NUMEXTFILES:=NUMEXTFILES+1;                                       90054000
              IF NUMEXTFILES<MAXEXTFILES THEN                                   90055000
              EXTFILETAB[NUMEXTFILES]:=CURNAME1 ELSE                            90056000
              IF NUMEXTFILES=MAXEXTFILES+1 THEN ERROR(73);                      90057000
            END;                                                                90058000
          END ELSE ERROR(9);                                                    90059000
          INSYMBOL;                                                             90060000
        END UNTIL CURSY#COMMA;                                                  90061000
        IF CURSY#RPAR THEN BEGIN ERROR(46); SKIP(SEMICOLON) END;                90062000
        IF CURSY=RPAR THEN INSYMBOL;                                            90063000
        IF CURSY#SEMICOLON THEN BEGIN ERROR(25); SKIP(SEMICOLON) END;           90064000
      END ELSE BEGIN ERROR(58); SKIP(SEMICOLON) END;                            90065000
    END ELSE BEGIN ERROR(9); SKIP(SEMICOLON) END;                               90066000
END ELSE BEGIN ERROR(75); SKIP(SEMICOLON) END;                                  90067000
INSYMBOL;                                                                       90068000
CURLEVEL:=1;                                                                    90069000
LASTREC:=MAXTABLES+1;                                                           90070000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%90071000
%                                                                               %90072000
          BLOCK;              %   COMPILE USER PROGRAM.                         %90073000
%                                                                               %90074000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%90075000
IF CURSY#DOT THEN                                                               90076000
BEGIN                                                                           90077000
    ERROR(76);                                                                  90078000
    DO BLOCK UNTIL CURSY=DOT;                                                   90079000
END;                                                                            90080000
IF FALSE THEN                                                                   90081000
BEGIN                                                                           90082000
ENDOFINPUT: ERROR(87);  CHARCNT:=-1;                                            90083000
            WRITE(LINES,TERMMESS);                                              90084000
END;                                                                            90085000
IF LISTOPTION AND CHARCNT>0 THEN PRINTLINE;                                     90086000
IF ERRINX>0 THEN PRINTERRORS;                                                   90087000
WRITE(LINES[DBL]);                                                              90088000
WRITE(LINES[DBL]);                                                              90089000
IF NUMERRS=0 THEN                                                               90090000
BEGIN                                                                           90091000
    ARRAY ZIPARRAY[0:191], Z[0:0];                                              90092000
    POINTER ZIPPNT;                                                             90093000
                                                                                90094000
    DEFINE ZIPTEXT(TEXT,L)=                                                     90095000
```

```
BEGIN                                                                   90096000
    Z[0]:=TEXT;                                                         90097000
    REPLACE ZIPPNT:ZIPPNT BY POINTER(Z[*])+(8-L) FOR L;                 90098000
END#;                                                                   90099000
                                                                        90100000
PROCEDURE ZIPNUM(N);    % TRANSFERS A NUMBER TO THE ZIP BUFFER.        90101000
VALUE N;  INTEGER N;                                                    90102000
IF N<9 THEN ZIPTEXT(N,1) ELSE                                           90103000
BEGIN ZIPNUM(N DIV 10); ZIPTEXT(ENTIER(N MOD 10),1) END;                90104000
                                                                        90105000
WRITEALGOL;                                                             90106000
WRITE(PASCALGOL,LASTLINE);                                              90107000
LOCK(PASCALGOL,SAVE);                                                   90108000
ZIPPNT:=POINTER(ZIPARRAY[*]);                                           90109000
REPLACE ZIPPNT BY " " FOR 20 WORDS;                                     90110000
WRITE(LINES,NOERRORS);                                                  90111000
ZIPTEXT("CC ",3);           ZIPTEXT("COMPILE",7);                       90112000
ZIPTEXT(" ",1);             ZIPTEXT(PROGNAME,PROGNAMELENGTH);           90113000
ZIPTEXT("/",1);             ZIPTEXT(USER,7);                            90114000
ZIPTEXT(" XALGOL",7);       ZIPTEXT(" ",1);                             90115000
IF SAVEFACTOR>0 THEN ZIPTEXT("LIBRARY",7);                              90116000
IF SAVEFACTOR<0 THEN ZIPTEXT("SYNTAX",6);                               90117000
ZIPTEXT(";",1);                                                         90118000
ZIPTEXT("XALGOL",6);        ZIPTEXT(" FILE",5);                         90119000
ZIPTEXT(" CARD=",6);        ZIPTEXT(ALGOLNAME,7);                       90120000
ZIPTEXT("/",1);             ZIPTEXT(USER,7);                            90121000
ZIPTEXT(" SERIAL",7);       ZIPTEXT(";",1);                             90122000
IF SAVEFACTOR>0 THEN                                                    90123000
BEGIN                                                                   90124000
  ZIPTEXT("SAVE=",5);  ZIPNUM(SAVEFACTOR);                              90125000
  ZIPTEXT(";",1);                                                       90126000
END;                                                                    90127000
ZIPTEXT("END.",4);                                                      90128000
ZIP WITH ZIPARRAY[*];                                                   90129000
END OF COMPILER ZIP FILE                                                90130000
BEGIN                                                                   91001000
    INTEGER I;                                                          91002000
    SWITCH FORMAT ERRORMESS:=                                           91003000
      (" 0  *** COMPILER ERROR *** CONTACT THE COMPUTER CENTRE."),      91004000
      (" 1  IDENTIFIER NOT DEFINED."),                                  91005000
      (" 2  IDENTIFIER ALREADY DEFINED."),                              91006000
      (" 3  WRONG NUMBER OF PARAMETERS."),                              91007000
      (" 4  SYNTAX ERROR."),                                            91008000
      (" 5  VARIABLE NOT ACCESSIBLE (HARDWARE RESTRICTION)."),          91009000
      (" 6  STRINGS MAY NOT BE CONTINUED FROM ONE CARD TO ANOTHER."),   91010000
      (" 7  A TYPE EXPECTED."),                                         91011000
      (" 8  VARIABLE EXPECTED."),                                       91012000
      (" 9  IDENTIFIER EXPECTED."),                                     91013000
      (" 10  INTEGER CONSTANT EXPECTED."),                              91014000
      (" 11  CONSTANT OF OTHER TYPE THAN EXPECTED."),                   91015000
      (" 12  VARIABLE OF ILLEGAL TYPE."),                               91016000
      (" 13  UNRECOGNIZABLE STATEMENT."),                               91017000
      (" 14  CONSTANT TOO BIG OR TOO SMALL."),                          91018000
      (" 15  UNDEFINED LABEL."),                                        91019000
      (" 16  FOR- AND CASE-STATEMENTS NESTED TOO DEEP."),               91020000
      (" 17  EXPRESSION IS OF WRONG TYPE."),                            91021000
      (" 18  """OF""" EXPECTED."),                                      91022000
```

```
(" 19  """DO""" EXPECTED."),                                                    91023000
(" 20  """ELSE""" WITHOUT CORRESPONDING """THEN""".."),                         91024000
(" 21  ILLEGAL TERMINATION OF STATEMENT."),                                     91025000
(" 22  """UNTIL""" EXPECTED."),                                                 91026000
(" 23  """TO"""/"""DOWNTO""" EXPECTED."),                                       91027000
(" 24  """END""" EXPECTED."),                                                   91028000
(" 25  """)""" EXPECTED."),                                                     91029000
(" 26  """:""" EXPECTED."),                                                     91030000
(" 27  """THEN""" EXPECTED."),                                                  91031000
(" 28  """:=""" EXPECTED."),                                                    91032000
(" 29  ONLY NUMBERS MAY BE SIGNED."),                                           91033000
(" 30  TOO MANY CASES."),                                                       91034000
(" 31  LABEL USED MORE THAN ONCE."),                                            91035000
(" 32  CONSTANT EXPECTED."),                                                    91036000
(" 33  LABEL NOT IN RANGE 0..9999."),                                           91037000
(" 34  TOO MANY LABELS DECLARED."),                                             91038000
(" 35  TOO MANY CONSTANTS DECLARED."),                                          91039000
(" 36  """=""" EXPECTED."),                                                     91040000
(" 37  THE LIST IS TOO LONG."),                                                 91041000
(" 38  INVALID TYPE FOR A FUNCTION."),                                          91042000
(" 39  """BEGIN""" EXPECTED."),                                                 91043000
(" 40  TOO MANY IDENTIFIERS DECLARED."),                                        91044000
(" 41  ALFA CONSTANS MAY NOT BE LONGER THAN 7 CHARACTERS."),                    91045000
(" 42  EXPRESSION IS NOT OF TYPE BOOLEAN."),                                    91046000
(" 43  NOT PROPER FORWARD DECLARATION."),                                       91047000
(" 44  UNSATISFIED FORWARD DECLARATION."),                                      91048000
(" 45  TOO MANY DIFFERENT TYPES DECLARED."),                                    91049000
(" 46  """)""" EXPECTED."),                                                     91050000
(" 47  """[""" EXPECTED."),                                                     91051000
(" 48  A SIMPLE TYPE EXPECTED."),                                               91052000
(" 49  """ARRAY OF ARRAY""" AND """ARRAY OF RECORD""" ILLEGAL",                 91053000
       " AS FILE TYPE."),                                                       91054000
(" 50  """FILE OF FILE""" IS ILLEGAL."),                                        91055000
(" 51  SET BOUNDRY IS TOO BIG OR TOO SMALL."),                                  91056000
(" 52  TOO MANY UNDECLARED POINTERS."),                                         91057000
(" 53  """..""" EXPECTED"),                                                     91058000
(" 54  FIRST VALUE IS GREATER THAN SECOND VALUE."),                             91059000
(" 55  TOO MANY RECORDS DECLARED AT ONE TIME."),                                91060000
(" 56  THE RECORD CONTAINS MORE THAN 1023 WORDS."),                             91061000
(" 57  FILES NOT ALLOWED IN RECORDS."),                                         91062000
(" 58  """(""" EXPECTED."),                                                     91063000
(" 59  """]""" EXPECTED.");                                                     91064000
                                                                                91065000
SWITCH FORMAT ERRORMESS2 :=                                                     91066000
(" 60  """ARRAY OF FILE""" NOT ALLOWED."),                                      91067000
(" 61  RANGE OF INDEX IS GREATER THAN 1023."),                                  91068000
(" 62  UNSATISFIED POINTER DECLARATION."),                                      91069000
(" 63  EXPRESSION IS TOO LONG."),                                               91070000
(" 64  ILLEGAL OPERATOR FOR THIS TYPE OF EXPRESSION."),                         91071000
(" 65  INTEGER EXPRESSION EXPECTED."),                                          91072000
(" 66  A SET EXPECTED."),                                                       91073000
(" 67  PARAMETER OF ILLEGAL TYPE."),                                            91074000
(" 68  PROCEDURES NOT ALLOWED IN THIS CONTEXT."),                               91075000
(" 69  ILLEGAL USE OF THIS TYPE OF IDENTIFIER."),                               91076000
(" 70  TOO MANY PARAMETERS DECLARED IN THE PROGRAM."),                          91077000
(" 71  """ARRAY OF CHAR""" EXPECTED."),                                         91078000
(" 72  WRONG TYPE OF SET EXPRESSION."),                                         91079000
```

```
("  73  TOO MANY EXTERNAL FILES."),                                        91080000
("  74  ILLEGAL IDENTIFIER FOR EXTERNAL FILE."),                           91081000
("  75  """PROGRAM""" EXPECTED."),                                        91082000
("  76  """."""" EXPECTED AT END OF PROGRAM."),                           91083000
("  77  EXTERNAL FILE IDENTIFIER MAY NOT EXCEED 6 CHARACTERS."),           91084000
("  78  ILLEGAL FILE PARAMETER."),                                         91085000
("  79  ILLEGAL USE OF FILE HANDLING PROCEDURE."),                         91086000
("  80  TEXT-FILE EXPECTED."),                                            91087000
("  81  POINTER VARIABLE EXPECTED."),                                      91088000
("  82  ONLY VALUES OF TYPE REAL, INTEGER OR CHAR MAY BE READ."),          91089000
("  83  VARIABLES IN RECORDS ILLEGAL IN THIS CONTEXT."),                   91090000
("  84  DISPLAY OVERFLOW."),                                              91091000
("  85  READ AND WRITE MAY ONLY BE USED ON TEXT-FILES."),                  91092000
("  86  REFERENCED OBJECT IS TOO BIG."),                                  91093000
("  87  END-OF-INPUT DISCOVERED."),                                       91094000
("  88  CHARACTER ARRAY EXPECTED."),                                       91095000
("  89  """,""" EXPECTED."),                                              91096000
("  90  PROCEDURES MAY NOT HAVE ANY TYPE."),                              91097000
("  91  PARAMETER OF WRONG KIND."),                                       91098000
("  92  ONLY COMPLETE ARRAYS AND RECORDS MAY BE TRANSMITTED."),           91099000
("  93  DECLARED LABEL NOT USED."),                                       91100000
("  94  PARAMETERS OF THIS TYPE SHOULD NOT BE VALUE PARAMETERS."),        91101000
("  95  ASSIGNMENT OF STRUCTURED VARIABLES NOT IMPLEMENTED."),           91102000
("  96  INPUT/OUTPUT NOT DECLARED."),                                    91103000
("  97 TOO MANY FILES IN USE."),                                         91104000
("  98  RECORD IDENTIFIER EXPECTED."),                                   91105000
("  99  UNRECOGNIZABLE ITEM."),                                          91106000
()};                                                                     91107000
                                                                         91108000
                                                                         91109000
    WRITE(LINES,ERRORS,NUMERRS);                                         91110000
    FOR I:=0 STEP 1 UNTIL 59 DO IF ERR[I] THEN                           91111000
    WRITE(LINES,ERRORMESS1[I]);                                          91112000
    FOR I:=60 STEP 1 UNTIL 119 DO IF ERR[I] THEN                         91113000
    WRITE(LINES,ERRORMESS2[I-60]);                                       91114000
END OF ERROR MESSAGES;                                                   91115000
IF XREFOPTION THEN                                                       92001000
BEGIN                                                                    92002000
    REPLACE POINTER(XREFLINE[*]) BY " " FOR 17 WORDS;                    92003000
    HEADING;                                                             92004000
    SORT(PRINTXREF,XREFFILE,0,XREFMAX,XREFCOMPARE,3,1000,6000);          92005000
END;                                                                     92006000
END OF B5700 PASCAL COMPILER.....                                        99001000
END;END.        LAST CARD IMAGE ON SOURCE TAPE FILE                      99999999
```

```
 LABEL  000000000LINE   001790A1CC EX 0/R;FILE LINE=LINE PRINT;COMMON=1;FILE S=SYMBOL/PASCAL;END←          0       /R
```

```
  50 PASCAL WAS REVISED 10/26/78 WITH NEW HERIOT-WATT VERSION.
 100 WE NOW HAVE THE HERIOT-WATT UNIVERSITY VERSION OF PASCAL.
 200 IT IS IMPLEMENTED AS A PRE-PROCESSOR THAT PRODUCES AN XALGOL
 300 PROGRAM FROM PASCAL SOURCE AND AUTOMATICALLY CALLS THE XALGOL
 400 COMPILER.
 500
 600 TO USE AS A BATCH COMPILER USE THE FOLLOWING CONTROL DECK.
 700 ?EXECUTE PASCAL/PASCAL
 800 ?DATA029 SOURCE   (THIS NAME IS LIKELY TO CHANGE)
 900    SOURCE DECK
1000 ?DATA029 INPUT
1100    INPUT DATA DECK IF ANY
1200 ?END
1300
1400 TO USE FROM TIME SHARING, FIRST CREATE SEQUENTIAL FILES OF THE
1500 SOURCE TEXT AND THE INPUT TEXT IF ANY. MAKE A FILE TO HOLD
1510  THE LISTING, BY MAKING A SEQUENTIAL FILE CONTAINING A LINE OF
1511  GARBAGE. (THE FILE MUST BE PRESENT BEFORE THE COMPILER CAN
1512  STORE INTO IT.)
1513
1514 THE SOURCE MUST INCLUDE A COMMENT  LINE (*$S10*)  THIS SETS
1515 THE SAVE FACTOR OPTION TO 10 DAYS, AND WILL CAUSE THE OBJECT
1516 FILE TO BE SAVED.  THE NAME IN THE "PROGRAM" STATEMENT SHOULD
1517 HAVE EXACTLY 6 LETTERS, AS THE SAVED VERSION WILL BE THAT
1518 NAME PREFIXED BY A ZERO, MAKING THE OBJECT PROGRAM AN EXECUTABLE
1519 PROGRAM IN YOUR ACCOUNT.
1520
1521   NEXT
1600      EQUATE CARD=<FILE>/<NAME> DISK SERIAL     FOR THE SOURCE
1700       EQUATE LINE=<FILE>/<NAME> DISK SERIAL    FOR THE LISTING
1710        (LISTING WILL GO TO LINE PRINTER IF YOU OMIT THIS)
1800      EXECUTE OBJECT PASCAL/PASCAL
1900
2000  THE REFERENCE MANUAL FOR PASCAL IS THE 1974 SPRINGER-VERLAG
2100 EDITION OF THE JENSEN & WIRTH REPORT.  THIS IS SUPPLEMENTED
2200 BY A HERIOT-WATT U. DOCUMENT.
2300 CURRENTLY AVAILABLE IS "PASCAL - USER MANUAL AND REPORT" BY JENSEN
2400 AND WIRTH, SPRINGER STUDY EDITION, AT $5.90 PLUS SALES TAX FROM
2500 SPRINGER-VERLAG NEW YORK, INC.,175 FIFTH AVE., NEW YORK, NY 10010
2600   (NEW YORK COLLECTS 6% SALES TAX FROM CALIFORNIANS.)
2700
2800 THE FOLLOWING MATERIAL IS TAKEN FROM THE HERIOT-WATT DOCUMENT BY
2900 DAG LANGMYHR, AUTHOR OF THE IMPLEMENTATION.
3000
3100 STRINGS MAY BE OF ANY LENGTH.  THE TYPE OF A STRING DEPENDS ON ITS
3200 LENGTH, THUS:
3300    LENGTH    TYPE
3400       0      ILLEGAL
3500       1      CHAR
3600     2-7      ALFA
3700    GTR 7     ILLEGAL, EXCEPT IN A WRITE STATEMENT
```