

?COMPILE JUNK/JUNK XALGOL SYNTAX

PACKET 1082
INPUT 5 CARDS FROM ZIP
TIME 1137
DATE 78087 TUESDAY, 03/28/78

*** BURROUGHS B5700 TSMCP MARK XVI.0.172 AND INTRINSICS MARK XVI.0.132 ***

#NO MESSAGES TODAY

11:37:53 ?COMPILE JUNK/JUNK XALGOL SYNTAX
11:37:56 ?XALGOL FILE TAPE=ELPASO/U819005 DISK SERIAL
11:37:57 ?DATA CARD
11:38:00 4:XALGOL/JUNK/HAYNES= 3 BOJ 1137 10/08/76
11:38:01 CDA IN CARD DA:XALGOL/JUNK= 3
11:38:02 DKA IN SER ELPASO U819005:XALGOL/JUNK= 3
11:38:03 -EOF NO LABEL CARD DA:XALGOL/JUNK= 3,NEAR LINE 2215750
11:38:04 CDA REL CARD DA:XALGOL/JUNK= 3
11:38:04 ?END
11:38:04 DKA REL ELPASO U819005:XALGOL/JUNK= 3
11:38:06 XALGOL/JUNK/HAYNES= 3,PST= 2,DS=ED
11:38:06 FOR XALGOL/JUNK= 3:PST= 2,IOT= 4,CORE=15360
11:38:07 PKT#1082 REMOVED

?COMPILE JUNK/JUNK XALGOL SYNTAX

PACKET 1080
INPUT 5 CARDS FROM ZIP
TIME 1131
DATE 78087 TUESDAY, 03/28/78

*** BURROUGHS B5700 TSMCP MARK XVI.0.172 AND INTRINSICS MARK XVI.0.132 ***

#NO MESSAGES TODAY

11:31:47 ?COMPILE JUNK/JUNK XALGOL SYNTAX
11:31:48 ?XALGOL FILE TAPE=ELPASO/U819005 DISK SERIAL
11:31:50 ?DATA CARD
11:31:53 4:XALGOL/JUNK/HAYNES= 3 BOJ 1131 10/08/76
11:31:55 CDA IN CARD DA:XALGOL/JUNK= 3
11:31:56 DKA IN SER ELPASO U819005:XALGOL/JUNK= 3
11:31:56 -EOF NO LABEL CARD DA:XALGOL/JUNK= 3, NEAR LINE 2215750
11:31:57 CDA REL CARD DA:XALGOL/JUNK= 3
11:31:57 ?END
11:31:57 DKA REL ELPASO U819005:XALGOL/JUNK= 3
11:31:59 XALGOL/JUNK/HAYNES= 3, PST= 2 DS=ED
11:32:00 FOR XALGOL/JUNK= 3: PST= 1, IOT= 3, CORE=15360
11:32:00 PKT#1080 REMOVED

```

$ CARD SEQXEQ RESET LIST%                                00001
%XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX% 00002
%                                                                 % 00003
%           THE PASCAL RUN TIME=SYSTEM,                  % 00004
%           -----                                         % 00005
%                                                                 % 00006
%XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX% 00007
BEGIN%                                                    00008
INTEGER V00167,V00168,V00169;%                            00009
FILE INPUT "INPUT" (2,10);%                               00010
FILE OUTPUT 1 (2,17);%                                   00011
%                                                         00012
DEFINE PRUCEDU =PROCEDURE#,%                             00013
    FUNCTN =REAL PROCEDURE#,%                            00014
    DOWNTD =STEP -1 UNTIL#,%                              00015
    UPTD   =STEP  1 UNTIL#,%                              00016
    B      =BOOLEAN#,%                                    00017
    F00603 =INPUT#,%                                     00018
    F00742 =OUTPUT#,%                                    00019
    LASTCH =[5:6]#,%                                     00020
    BUFSIZE =[13:8]#,%                                   00021
    BUFPNT  =[21:8]#,%                                   00022
    EOF     =[22:1]#,%                                   00023
    EOLN    =[23:1]#,%                                   00024
    INP     =[24:1]#,%                                   00025
    OUTP    =[25:1]#,%                                   00026
    ENDFOUND=[26:1]#,%                                   00027
    MEMSIZE =10000#,%                                    00028
    MAXINT  =549755813887#;%                             00029
%                                                         00030
ARRAY MEM[0:MEMSIZE DIV 1022,0:1022], TEXT,CHAR[0:0], TEMPTXT[0:19],% 00031
    V00603[0:9], V00742[0:16];%                          00032
INTEGER MEMPNT,T,T1,I00603,I00742;%                      00033
POINTER CHARPNT,TEXTPNT;%                                00034
LABEL TERMINATE;%                                       00035
FORMAT TERMMESS ("**** PROGRAM EXECUTION TERMINATED AT LINE ",I*,"."),% 00036
    CHECKERR ("**** THE VALUE ",I*," IS NOT IN THE RANGE ",I*,".",% 00037
    I*,"."),%                                           00038
    ERRMARK (X*,"X"),%                                    00039
    CONCATERR("**** CONCAT ERROR: [",I*,";",I*,";",I*,"]"),% 00040
    ILLEGALCC("**** ILLEGAL CARRIAGE CONTROL CHARACTER:""",A1,"");% 00041
SWITCH FORMAT ERRMESS :=%                                 00042
    (%),%                                                 00043
    ("**** NO READING WHILE EOF IS TRUE,"),              %1 00044
    ("**** NO WRITING WHILE EOF IS FALSE,"),            %2 00045
    ("**** ILLEGAL CHARACTER,"),                        %3 00046
    ("**** OVERFLOW ERROR,"),                           %4 00047
    ("**** NO RESET/REWRITE ON INPUT/OUTPUT,"),        %5 00048
    ("**** LINE IMAGE OVERFLOW,");                      %6 00049
MONITOR EXPBVR:=REALOVERFLOW;%                           00050
%                                                         00051
INTEGER PROCEDURE NUMDIGITS(N);%                         00052
VALUE N; INTEGER N;%                                     00053

```

NUMDIGITS:=IF N<0 THEN 1+NUMDIGITS(-N) ELSE%	00054
IF N>9 THEN 1+NUMDIGITS(N DIV 10) ELSE 1;%	00055
%	00056
PROCEDURE RUNERR(ERRNUM,LINENUM);	00057
VALUE ERRNUM,LINENUM;%	00058
INTEGER ERRNUM,LINENUM;%	00059
BEGIN%	00060
WRITE(OUTPUT,ERRMESS[ERRNUM]);%	00061
WRITE(OUTPUT,TERMMESS,NUMDIGITS(LINENUM),LINENUM);%	00062
GO TO TERMINATE;%	00063
END OF RUNERR;%	00064
%	00065
INTEGER PROCEDURE CHECK(VAL,LIM1,LIM2,LINENUM);%	00066
VALUE VAL,LIM1,LIM2,LINENUM;%	00067
INTEGER VAL,LIM1,LIM2,LINENUM;%	00068
BEGIN%	00069
IF VAL<LIM1 OR VAL>LIM2 THEN%	00070
BEGIN WRITE(OUTPUT,CHECKERR,NUMDIGITS(VAL),VAL,NUMDIGITS(LIM1),%	00071
LIM1,NUMDIGITS(LIM2),LIM2);%	00072
RUNERR(4,LINENUM);%	00073
END;%	00074
CHECK:=VAL;%	00075
END OF CHECK;%	00076
%	00077
ALPHA PROCEDURE CURDAT;%	00078
CURDAT:=" "&TIME(5)[41:35:36];%	00079
%	00080
ALPHA PROCEDURE WEEKDA;%	00081
WEEKDA:=TIME(6)&" "[41:5:6];%	00082
%	00083
INTEGER PROCEDURE TRUNC(X,LINENUM);%	00084
VALUE X,LINENUM;%	00085
REAL X; INTEGER LINENUM;%	00086
BEGIN%	00087
IF ABS(X)>MAXINT THEN RUNERR(4,LINENUM);%	00088
TRUNC:=IF X<0 THEN -ENTIER(-X) ELSE ENTIER(X);%	00089
END OF TRUNC;%	00090
%	00091
INTEGER PROCEDURE ROUND(X,LINENUM);%	00092
VALUE X,LINENUM;%	00093
REAL X; INTEGER LINENUM;%	00094
BEGIN%	00095
IF ABS(X)>MAXINT THEN RUNERR(4,LINENUM);%	00096
ROUND:=X;%	00097
END OF ROUND;%	00098
%	00099
BOOLEAN PROCEDURE ODD(N);%	00100
VALUE N; INTEGER N;%	00101
ODD:=N MOD 2 = 1;%	00102
%	00103
REAL PROCEDURE SQR(X,LINENUM);%	00104
VALUE X,LINENUM;%	00105
REAL X; INTEGER LINENUM;%	00106
BEGIN%	00107
IF ABS(X)>2.0769187@34 THEN RUNERR(4,LINENUM);%	00108
SQR:=X*X;%	00109
END OF SQR;%	00110

```

%
BOOLEAN PROCEDURE INCL1(A,B);          %*** IS THE SET "A" INCLUDED      00111
VALUE A,B; REAL A,B;                  %*** IN THE SET "B",          00112
INCL1:=REAL(BOOLEAN(A) AND NOT BOOLEAN(B))=0;%      00113
%
BOOLEAN PROCEDURE INCL2(A,B);          %*** IS THE SET "B" INCLUDED      00114
VALUE A,B; REAL A,B;                  %*** IN THE SET "A",          00115
INCL2:=REAL(BOOLEAN(B) AND NOT BOOLEAN(A))=0;%      00116
%
BOOLEAN PROCEDURE INTST(A,B);          %*** IS THE VALUE "A" AN ELEMENT 00117
VALUE A,B; REAL A,B;                  %*** IN THE SET "B",          00118
INTST:=IF A<0 OR A>38 THEN FALSE ELSE O&B[0:38-A:1]=1;% 00119
%
PROCEDURE NEW(P,SIZE);%
VALUE SIZE; REAL P; INTEGER SIZE;%     00120
BEGIN%                                  00121
  P:=IF MEMPNT+SIZE>MEMSIZE THEN 0 ELSE MEMPNT;% 00122
  MEMPNT:=MEMPNT+SIZE;%                00123
END OF NEW;%                             00124
%
PROCEDURE DISPOSE(P,SIZE);%
VALUE SIZE; REAL P; INTEGER SIZE;%     00125
BEGIN%                                  00126
END OF DISPOSE;%                         00127
%
PROCEDURE PACK(A,LLIM,ULIM,I,Z,LINENUM);% 00128
VALUE LLIM,ULIM,I,LINENUM;%           00129
ARRAY A[*]; ALPHA Z;%                 00130
INTEGER LLIM,ULIM,I,LINENUM;%         00131
BEGIN%                                  00132
  Z:=0;%                                00133
  FOR T1:=0 STEP 1 UNTIL 6 DO%          00134
    Z:=A[CHECK(I+T1,LLIM,ULIM,LINENUM)] & Z [41:35:36];% 00135
  END;%                                  00136
%
PROCEDURE UNPACK(Z,A,LLIM,ULIM,I,LINENUM);% 00137
VALUE Z,LLIM,ULIM,I,LINENUM;%         00138
ARRAY A[*]; ALPHA Z;%                 00139
INTEGER LLIM,ULIM,I,LINENUM;%         00140
FOR T1:=0 STEP 1 UNTIL 6 DO%           00141
  A[CHECK(I+T1,LLIM,ULIM,LINENUM)]:= 0 & Z [5:41-6*T1:6];% 00142
%
REAL PROCEDURE CONCAT(A,B,AS,BS,N,LINENUM);% 00143
VALUE A,B,AS,BS,N,LINENUM;%           00144
REAL A,B; INTEGER AS,BS,N,LINENUM;%   00145
BEGIN%                                  00146
  IF AS<1 OR BS<1 OR N<0 OR AS+N>48 OR BS+N>48 THEN% 00147
    BEGIN%                                00148
      WRITE(OUTPUT,CONCATERR,NUMDIGITS(AS),AS,NUMDIGITS(BS),% 00149
        BS,NUMDIGITS(N),N);%              00150
      RUNERR(0,LINENUM);%                00151
    END;%                                  00152
    CONCAT:=A & B [47-AS:47-BS:N];%      00153
  END OF CONCAT;%                         00154
%
BOOLEAN PROCEDURE BIT(N,LINENUM);      %*** SET BIT NO "N" IN A WORD. 00155
VALUE N,LINENUM; INTEGER N,LINENUM;%   00156

```

```

00111
00112
00113
00114
00115
00116
00117
00118
00119
00120
00121
00122
00123
00124
00125
00126
00127
00128
00129
00130
00131
00132
00133
00134
00135
00136
00137
00138
00139
00140
00141
00142
00143
00144
00145
00146
00147
00148
00149
00150
00151
00152
00153
00154
00155
00156
00157
00158
00159
00160
00161
00162
00163
00164
00165
00166
00167

```

```

BIT:=BOOLEAN(0 & 1 [38-CHECK(N,0,38,LINENUM):0:1]);% 00168
% 00169
BOOLEAN PROCEDURE BITS(N1,N2,LINENUM); %*** SET BITS "N1"..,"N2". 00170
VALUE N1,N2,LINENUM;% 00171
INTEGER N1,N2,LINENUM;% 00172
BITS:=BOOLEAN(0 & $"777777777777" [38-CHECK(N1,0,38,LINENUM):38;% 00173
CHECK(N2,0,38,LINENUM)-N1+1]);% 00174
% 00175
PROCEDURE RLINE(F,BUF,INFO);% 00176
FILE F; ARRAY BUF[0]; INTEGER INFO;% 00177
BEGIN% 00178
  LABEL ENDFILE;% 00179
  INFO.EOLN:=0; INFO.BUFPNT:=1;% 00180
  READ(F,999,BUF[*]) [ENDFILE];% 00181
  REPLACE CHARPNT BY POINTER(BUF[*]) FOR 1;% 00182
  INFO.LASTCH:=CHAR[0];% 00183
  IF FALSE THEN% 00184
  BEGIN ENDFILE: INFO.ENDFOUND:=1;% 00185
  END;% 00186
END OF RLINE;% 00187
% 00188
REAL PROCEDURE PREAD(F,BUF,INFO,MODE,LINENUM);% 00189
VALUE MODE,LINENUM;% 00190
FILE F; ARRAY BUF[0];% 00191
INTEGER INFO,MODE,LINENUM;% 00192
BEGIN% 00193
  DEFINE GETCHAR=% 00194
  BEGIN% 00195
    IF BOOLEAN(INFO,EOLN) THEN% 00196
    BEGIN% 00197
      REINE(F,BUF,INFO); CH:=INFO.LASTCH;% 00198
    END ELSE% 00199
    IF INFO.BUFPNT=INFO.BUFSIZE THEN% 00200
    BEGIN CH:=" "; INFO.EOLN:=1 END ELSE% 00201
    BEGIN% 00202
      REPLACE CHARPNT BY POINTER(BUF[*])+INFO.BUFPNT FOR 1;% 00203
      CH:=CHAR[0]; INFO.BUFPNT:=INFO.BUFPNT+1;% 00204
    END END OF GETCHAR;% 00205
% 00206
  DEFINE READERR(ERRNUM)=% 00207
  BEGIN% 00208
    WRITE(OUTPUT,999,BUF[*]);% 00209
    WRITE(OUTPUT,ERRMARK,INFO.BUFPNT-1);% 00210
    RUNERR(ERRNUM,LINENUM);% 00211
  END OF READERR;% 00212
% 00213
  REAL RES; ALPHA CH;% 00214
  BOOLEAN NEGATIVE,NEGEXP; INTEGER POWER,EXP;% 00215
  LABEL OVERFLOW,RETURN;% 00216
% 00217
  IF BOOLEAN(INFO,EOF) THEN RUNERR(1,LINENUM);% 00218
  IF BOOLEAN(INFO,ENDFOUND) THEN% 00219
  BEGIN% 00220
    INFO.EOF:=1; PREAD:=0;% 00221
    GO TO RETURN;% 00222
  END;% 00223
  IF MODE=1 THEN %*** MODE = CHAR *** 00224

```

```

BEGIN%
  PREAD:=INFO.LASTCH; GETCHAR; INFO.LASTCH:=CH;%
END ELSE%
BEGIN
  CH:=INFO.LASTCH;%
  WHILE CH=" " AND NOT BOOLEAN(INFO.ENDFOUND) DO GETCHAR;%
  IF BOOLEAN(INFO.ENDFOUND) THEN%
  BEGIN%
    INFO.EOF:=1; PREAD:=0;%
    GO TO RETURN;%
  END;%
  IF CH="+" OR CH="-" THEN BEGIN NEGATIVE:=CH="-"; GETCHAR END;%
  IF CH>9 THEN READERR(3);%
  RES:=CH; GETCHAR;%
  WHILE CH<9 DO BEGIN RES:=10*RES+CH; GETCHAR END;%
  IF MODE=3 THEN % MODE = REAL,
  BEGIN%
    IF CH="." THEN%
    BEGIN%
      GETCHAR; IF CH>9 THEN READERR(3);%
      WHILE CH<9 DO BEGIN RES:=10*RES+CH;POWER:=POWER-1;GETCHAR END;%
    END;%
    IF CH="E" THEN%
    BEGIN%
      GETCHAR;%
      IF CH="+" OR CH="-" THEN BEGIN NEGEXP:=CH="-"; GETCHAR END;%
      IF CH>9 THEN READERR(3);%
      WHILE CH<9 DO BEGIN EXP:=10*EXP+CH; GETCHAR END;%
      IF NEGEXP THEN EXP:=-EXP;%
    END;%
    POWER:=POWER+EXP;%
    REALOVERFLOW:=OVERFLOW; RES:=RES*10*POWER;%
    IF FALSE THEN OVERFLOW; READERR(4);%
    REALOVERFLOW:=0;%
    END ELSE IF RES>MAXINT THEN READERR(4);%
    PREAD:=IF NEGATIVE THEN -RES ELSE RES;%
    INFO.EASTCH:=CH;%
  END;%
RETURN;%
END OF PREAD;%
%
%
PROCEDURE WLINE(F,BUF,INFO); %*** PRINT A LINE.***
FILE F; ARRAY BUF(0); INTEGER INFO;%
BEGIN%
  ALPHA CC;%
  IF BOOLEAN(INFO.OUTPUT) THEN%
  BEGIN%
    REPLACE CHARPNT BY POINTER(BUF[*]) FOR 1; CC:=CHAR(0);%
    REPLACE POINTER(BUF[*]) BY " ";%
    IF CC=" " THEN WRITE(OUTPUT,999,BUF[*]) ELSE%
    IF CC="+" THEN WRITE(OUTPUT[NO],999,BUF[*]) ELSE%
  BEGIN%
    IF CC="0" THEN WRITE(OUTPUT) ELSE%
    IF CC="-" THEN WRITE(OUTPUT[DBL]) ELSE%
    IF CC="1" THEN WRITE(OUTPUT[PAGE]) ELSE%
    WRITE(OUTPUT,ILLEGALCC,CC);%
  END%
  END%

```

```

00225
00226
00227
00228
00229
00230
00231
00232
00233
00234
00235
00236
00237
00238
00239
00240
00241
00242
00243
00244
00245
00246
00247
00248
00249
00250
00251
00252
00253
00254
00255
00256
00257
00258
00259
00260
00261
00262
00263
00264
00265
00266
00267
00268
00269
00270
00271
00272
00273
00274
00275
00276
00277
00278
00279
00280
00281

```

```

        WRITE(OUTPUT,999,BUF[*]);%
    END;%
    END ELSE WRITE(F,999,BUF[*]);%
    REPLACE POINTER(BUF[*]) BY " " FOR INFO,BUFSIZE;%
    INFO,BUFPNT:=0;%
END OF WLINE;%
%
%
PROCEDURE CHFIL(F);%
FILE F;%
BEGIN%
    ARRAY A[0:6];%
    SEARCH(F,A[*]);%
    IF A[0]=-1 THEN%
    BEGIN%
        F.AREAS := 20;%
        F.AREASIZE := 300;%
    END;%
END OF CHFIL;%
%
%
PROCEDURE WALFA(F,BUF,INFO,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12,ALENG,
                LINENUM);%
VALUE A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12,ALENG,LINENUM;%
FILE F; ARRAY BUF[0]; INTEGER INFO,ALENG,LINENUM;%
ALPHA A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12;%
BEGIN%
    ALPHA A; POINTER PNT;%
    LABEL EXIT;%
    IF NOT BOOLEAN(INFO.EOF) THEN RUNERR(2,LINENUM);%
    IF INFO.BUFPNT+ALENG>INFO.BUFSIZE THEN WLINE(F,BUF,INFO);%
    PNT:=POINTER(BUF[*])+INFO.BUFPNT;%
    INFO.BUFPNT:=INFO.BUFPNT+ALENG;%
    FOR A:=A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12 DO%
    BEGIN%
        TEXT[0]:=A;%
        REPLACE PNT:PNT BY TEXTPNT FOR MIN(ALENG,7);%
        ALENG:=ALENG-7; IF ALENG<=0 THEN GO TO EXIT;%
    END;%
EXIT;%
END OF WALFA;%
%
%
PROCEDURE PWRITE(F,BUF,INFO,E,EMODE,M,N,LINENUM);%
VALUE E,EMODE,M,N,LINENUM;%
FILE F; ARRAY BUF[0]; REAL E;%
INTEGER INFO,EMODE,M,N,LINENUM;%
BEGIN%
    INTEGER NCHARS,NEXP,I; POINTER CPNT;%
    DEFINE PUTCHAR(C)= % PUTS A CHARACTER INTO TEMPTXT
    BEGIN CHAR[0]:=C; NCHARS:=NCHARS+1;%
        REPLACE CPNT:CPNT BY CHARPNT FOR 1;%
    END#;%
%
%
PROCEDURE PUTINT(N); % PUTS AN INTEGER INTO TEMPTXT
VALUE N; INTEGER N; % WITH ZERO SUPPRESSION,
IF N<9 THEN PUTCHAR(N) ELSE%

```

```

00282
00283
00284
00285
00286
00287
00288
00289
00290
00291
00292
00293
00294
00295
00296
00297
00298
00299
00300
00301
00302
00303
00304
00305
00306
00307
00308
00309
00310
00311
00312
00313
00314
00315
00316
00317
00318
00319
00320
00321
00322
00323
00324
00325
00326
00327
00328
00329
00330
00331
00332
00333
00334
00335
00336
00337
00338

```



```

BEGIN PUTINT(N DIV 10); PUTCHAR(ENTIER(N MOD 10)) END;% 00339
* CPNT:=POINTER(TEMPTTEXT[*]);% 00340
IF NOT BOOLEAN(INFO.EOF) THEN RUNERR(2,LINENUM);% 00341
IF EMODE=1 THEN %*** MODE = INTEGER *** 00342
BEGIN% 00343
IF E<0 THEN BEGIN PUTCHAR("-"); E:=-E END;% 00344
PUTINT(E);% 00345
END ELSE% 00346
IF EMODE=2 THEN %*** MODE = REAL *** 00347
BEGIN% 00348
PUTCHAR(" ");% 00349
IF E<0 THEN BEGIN PUTCHAR("-"); E:=-E END;% 00350
IF E>MAXINT OR N<0 THEN % FLOATING-POINT. 00351
BEGIN% 00352
IF E>0 THEN% 00353
BEGIN% 00354
WHILE E<1 DO BEGIN NEXP:=NEXP-1; E:=10*E END;% 00355
WHILE E≥10 DO BEGIN NEXP:=NEXP+1; E:=E/10 END;% 00356
END;% 00357
I:=MAX(M-8,1);% 00358
E:=E+0.5*10*(-I);% 00359
IF E GE 10 THEN BEGIN NEXP:=NEXP+1; E:=E/10 END;% 00360
PUTCHAR(ENTIER(E)); E:=E-ENTIER(E); PUTCHAR(".");% 00361
DO BEGIN 00362
E:=10*E; PUTCHAR(ENTIER(E));% 00363
E:=E-ENTIER(E); I:=I-1;% 00364
END UNTIL I≤0;% 00365
PUTCHAR("E");% 00366
IF NEXP<0 THEN BEGIN PUTCHAR("-"); NEXP:=-NEXP END% 00367
ELSE PUTCHAR("+");% 00368
PUTCHAR(NEXP DIV 10); PUTCHAR(ENTIER(NEXP MOD 10));% 00369
END ELSE% 00370
BEGIN % FIXED-POINT. 00371
E:=E+0.5*10*(-N);% 00372
PUTINT(ENTIER(E)); PUTCHAR("."); E:=E-ENTIER(E);% 00373
IF N>150 THEN RUNERR(6,LINENUM);% 00374
FOR I:=1 STEP 1 UNTIL N DO% 00375
BEGIN E:=10*E; PUTCHAR(ENTIER(E));% 00376
E:=E-ENTIER(E);% 00377
END END ELSE% 00378
IF EMODE=3 THEN %*** MODE = BOOLEAN *** 00379
BEGIN% 00380
IF E<0.5 THEN REPLACE CPNT BY "FALSE" ELSE REPLACE CPNT BY "TRUE"; 00381
NCHARS:=IF E<0.5 THEN 5 ELSE 4;% 00382
END ELSE% 00383
IF EMODE=5 THEN %*** MODE = ALFA *** 00384
BEGIN% 00385
TEXT[0]:=E; NCHARS:=MIN(M,7);% 00386
REPLACE CPNT:CPNT BY TEXTPNT FOR 7;% 00387
END ELSE% 00388
BEGIN %*** MODE = CHAR *** 00389
PUTCHAR(E);% 00390
END;% 00391
IF NCHARS>M THEN M:=NCHARS;% 00392
IF INFO.BUFPNT+M>INFO.BUFSIZE THEN WLINE(F,BUF,INFO);% 00393
IF M>INFO.BUFSIZE THEN RUNERR(6,LINENUM);% 00394
00395

```

REPLACE POINTER(BUF[*])+(INFO.BUFPNT+M-NCHARS) BY%	00396
POINTER(TEMPTEXT[*]) FOR NCHARS;%	00397
INFO.BUFPNT:=INFO.BUFPNT+M;%	00398
END OF PWRITE;%	00399
%	00400
%	00401
PROCEDURE PUT(F,BUF,INFO,LINENUM);%	00402
VALUE LINENUM;%	00403
FILE F; ARRAY BUF[*];%	00404
INTEGER INFO,LINENUM;%	00405
BEGIN%	00406
IF INFO.BUFSIZE=0 THEN%	00407
BEGIN%	00408
IF NOT BOOLEAN(INFO.EOF) THEN RUNERR(2,LINENUM);%	00409
WRITE(F,1023,BUF[*]);%	00410
END ELSE PWRITE(F,BUF,INFO,INFO,LASTCH,4,1,1,LINENUM);%	00411
END OF PUT;%	00412
%	00413
%	00414
PROCEDURE GET(F,BUF,INFO,LINENUM);%	00415
VALUE LINENUM;%	00416
FILE F; ARRAY BUF[*];%	00417
INTEGER INFO,LINENUM;%	00418
BEGIN%	00419
ALPHA X; LABEL ENDFILE;%	00420
IF INFO.BUFSIZE=0 THEN%	00421
BEGIN%	00422
IF BOOLEAN(INFO.EOF) THEN RUNERR(1,LINENUM);%	00423
READ(F,1023,BUF[*]) [ENDFILE];%	00424
IF FALSE THEN ENDFILE; INFO.EOF:=1;%	00425
END ELSE X:=PREAD(F,BUF,INFO,1,LINENUM);%	00426
END OF GET;%	00427
%	00428
%	00429
PROCEDURE PPAGE(F,BUF,INFO,LINENUM);%	00430
VALUE LINENUM;%	00431
FILE F; ARRAY BUF[*];%	00432
INTEGER INFO,LINENUM;%	00433
BEGIN%	00434
IF NOT BOOLEAN(INFO.EOF) THEN RUNERR(2,LINENUM);%	00435
WRITE(F,PPAGE);%	00436
END OF PPAGE;%	00437
%	00438
%	00439
PROCEDURE RESET(F,BUF,INFO,LINENUM);%	00440
VALUE LINENUM;%	00441
FILE F; ARRAY BUF[*];%	00442
INTEGER INFO,LINENUM;%	00443
BEGIN%	00444
IF BOOLEAN(INFO.INP) OR BOOLEAN(INFO.OUTP) THEN RUNERR(5,LINENUM);%	00445
REWIND(F); INFO.EOF:=0; INFO.EOLN:=0; INFO.BUFPNT:=0;%	00446
INFO.ENDFOUND:=0;%	00447
IF INFO.BUFSIZE=0 THEN GET(F,BUF,INFO,LINENUM);%	00448
ELSE RLINE(F,BUF,INFO);%	00449
END OF RESET;%	00450
%	00451
PROCEDURE REWRITE(F,BUF,INFO,LINENUM);%	00452

VALUE LINENUM;%	00453
FILE F; ARRAY BUF[*];%	00454
INTEGER INFO,LINENUM;%	00455
BEGIN%	00456
IF BOOLEAN(INFO,INP) OR BOOLEAN(INFO,OUTP) THEN RUNERR(5,LINENUM);%	00457
REWIND(F); INFO.EOF:=1; INFO.BUFPNT:=0; INFO.ENDFOUND:=0;%	00458
IF INFO.BUFSIZE>0 THEN%	00459
REPLACE POINTER(BUF[*]) BY " " FOR INFO.BUFSIZE;%	00460
END OF REWRITE;%	00461
%	00462
%	00463
PROCEDURE INIT(INPUTDECL);%	00464
VALUE INPUTDECL;%	00465
BOOLEAN INPUTDECL;%	00466
BEGIN%	00467
MEMPNT:=1;%	00468
CHARPNT:=POINTER(CHAR[*])+7; TEXTPNT:=POINTER(TEXT[*])+1;%	00469
T:=0; T.BUFSIZE:=80; T.BUFPNT:=80; T.EOLN:=1; T.INP:=1;%	00470
I00603:=T; IF INPUTDECL THEN RLINE(INPUT,V00603,I00603);%	00471
T:=0; T.BUFSIZE:=132; T.EOLN:=1; T.OUTP:=1; T.EOF:=1;%	00472
I00742:=T;%	00473
REPLACE POINTER(V00742[*]) BY " " FOR 17 WORDS;%	00474
END OF INIT;%	00475

LABEL 00000000 LINE 00178087CC EX 0/R;FILE S=PASC RUN/DISK;FILE LINE=LINE PRINT;COMMON=1;END+

0 /R

?COMPILE JUNK/JUNK XALGOL SYNTAX

PACKET 1084 CARDS FROM ZIP
TIME 1139
DATE 78087 TUESDAY, 03/28/78

*** BURROUGHS B5700 TSMCP MARK XVI.0.172 AND INTRINSICS MARK XVI.0.132 ***

#NO MESSAGES TODAY

```
11:39:17 ?COMPILE JUNK/JUNK XALGOL SYNTAX
11:39:20 ?XALGOL FILE TAPE=ELPASO/HAYNES DISK SERIAL 0
11:39:20 ?DATA CARD
11:39:24 4:XALGOL/JUNK/HAYNES= 3 BDJ 1139 10/08/76
11:39:26 CDA IN CARD DA:XALGOL/JUNK= 3
11:39:28 DKA IN SER ELPASO HAYNES:XALGOL/JUNK= 3
11:39:30 PBD1085 OUT 011 LINE:XALGOL/JUNK= 3
11:39:33 DKA OUT SER DSK1 HAYNES:XALGOL/JUNK= 3
11:39:57 DKA OUT RDM JUNK JUNK:XALGOL/JUNK= 3
11:40:01 DKA OUT SER DSK2 HAYNES:XALGOL/JUNK= 3
11:48:30 3TI
11:48:30 TIME FOR XALGOL/JUNK= 3 IS 4:26 IN 9:11
11:49:38 PBD1085 OUT 012:XALGOL/JUNK= 3
11:51:07 CDA REL CARD DA:XALGOL/JUNK= 3
11:51:07 ?END
11:51:07 DKA REL ELPASO HAYNES:XALGOL/JUNK= 3
11:51:08 DKA OUT SER DSRT1 HAYNES:XALGOL/JUNK= 3
11:51:20 DKA OUT SER DSRT2 HAYNES:XALGOL/JUNK= 3
11:51:30 DSRT1/HAYNES/HAYNES= 0 SEGS--CREATED 03/28/78 AT 11:51:07:18
11:51:30 DKA REL DSRT1 HAYNES:XALGOL/JUNK= 3
11:51:30 DSRT2/HAYNES/HAYNES= 0 SEGS--CREATED 03/28/78 AT 11:51:18:53
11:51:31 DKA REL DSRT2 HAYNES:XALGOL/JUNK= 3
11:51:31 DKA OUT SER DSRT1 HAYNES:XALGOL/JUNK= 3
11:53:29 DKA OUT SER DSRT2 HAYNES:XALGOL/JUNK= 3
11:55:10 #NO USER DISK:XALGOL/JUNK= 3= 151 SEGS.
11:55:39 3OK
11:55:45 #NO USER DISK:XALGOL/JUNK= 3= 151 SEGS.
11:56:13 3OK
11:56:22 #NO USER DISK:XALGOL/JUNK= 3= 151 SEGS.
11:56:27 DS OK
11:56:27 #NO USER DISK:XALGOL/JUNK= 3= 151 SEGS.
11:58:46 DS OK
11:58:46 #NO USER DISK:XALGOL/JUNK= 3= 151 SEGS.
12:00:08 3DS
12:00:11 -OPRTR DS=ED:XALGOL/JUNK= 3, NEAR LINE 17100100
12:00:11 DSK2/HAYNES/HAYNES= 900 SEGS--CREATED 03/28/78 AT 11:40:00:13
12:00:11 DKA REL DSK2 HAYNES:XALGOL/JUNK= 3
12:00:12 DSK1/HAYNES/HAYNES= 300 SEGS--CREATED 03/28/78 AT 11:39:31:39
12:00:12 DKA REL DSK1 HAYNES:XALGOL/JUNK= 3
12:00:12 PBD1085 REL 012 LINE 7502:XALGOL/JUNK= 3
12:00:12 JUNK/JUNK/HAYNES= 630 SEGS--CREATED 03/28/78 AT 11:39:55:41
```

12:00:13
12:00:13
12:00:13
12:00:13
12:00:14
12:00:16
12:00:16
12:00:16

DKA REL JUNK JUNK:XALGOL/JUNK= 3
DSRT2/HAYNES/HAYNES= 2112 SEGS==CREATED 03/28/78 AT 11:53:28:41
DKA REL DSRT2 HAYNES:XALGOL/JUNK= 3
DSRT1/HAYNES/HAYNES= 2112 SEGS==CREATED 03/28/78 AT 11:51:30:18
DKA REL DSRT1 HAYNES:XALGOL/JUNK= 3
XALGOL/JUNK/HAYNES= 3,PST= 10:05 DS=ED
FOR XALGOL/JUNK= 3:PST= 605,IOT= 273,CORE=15360
PKT#1084 REMOVED


```

MAXCONSTS =200#,%      %SIZE OF CONSTANT TABLE,
MAXTEMPS  =5#,%        %NUMBER OF EXTRA VARS IN EACH PROCEDURE,
MAXWITHSYMS=250#,%    %MAX NUMBER OF SYMBOLS USED BY WITH=STATMS,
MAXSYMS   =800#,%     %MAX NUMBER OF SYMBOLS IN ONE EXPRESSION,
LISTLENGTH=800#,%    %MAX LENGTH OF VAR AND PARAM LISTS,
MAXEXTFILES=20#,%    %MAX NUMBER OF EXTERNAL FILES,
MAXFILES  =20#,%     %MAX NUMBER OF FILES DECLARED AT ONE TIME,
MAXPNTRS  =50#,%     %MAX NUMBER OF UNDECLARED POINTERS.

%
%*** NAME TABLES ***
ARRAY NAMETAB1,NAMETAB2,NAMETAB3[0:MAXTABLES,0:MAXNAMES];%
DEFINE NAMELENGTH =[41:6]#,%
      TYPE          =[9:10]#,%
      IDCLASS       =[12:3]#,%
      VAR           =0#,%
      CONST         =1#,%
      FUNC          =2#,%
      PROC          =3#,%
      TYPES         =4#,%
      INFO          =[23:11]#,%
      FORMAL        =[24:1]#,%
      FORWARDDEF    =[25:1]#,%
      EXTERNALFILE=[26:1]#;%

%
%*** DISPLAY VECTOR ***
ARRAY DISPLAY[0:MAXLEVEL];%
DEFINE RECTYPE      =[9:10]#,%
      FIRSTWITHSYM  =[19:10]#,%
      LASTWITHSYM   =[29:10]#,%
      NUMPNTRSINWITH=[35:6]#,%
      BRACKETSINWITH=[36:1]#,%
      NAMETAB       =[46:7]#;%

%
%*** TYPE TABLES ***
ARRAY TYPETAB1,TYPETAB2,TYPETAB3[0:MAXTYPES];%
DEFINE FORM         =[3:4]#,%
      NUMERIC       =0#,%
      SYMBOLIC      =1#,%
      SUBTYPE       =2#,%
      MAINTYPE      =[33:10]#,%
      CHAR          =3#,%
      FLOATING      =4#,%
      ALFA          =5#,%
      SET           =6#,%
      SETTYPE       =[33:10]#,%
      POINTERS      =7#,%
      PQINTTYPE     =[33:10]#,%
      ARRAYS        =8#,%
      INXTYPE       =[33:10]#,%
      ARRTYPE       =[43:10]#,%
      RECORD        =9#,%
      RECTAB        =[33:10]#,%
      FILES         =10#,%
      FILETYPE      =[33:10]#,%
      TEXTFILE      =11#,%
      SIZE          =[15:12]#,%
      STRUCT        =[23:8]#;%

```

```

00049 T 0019
00050 T 0013
00051 T 0013
00052 T 0013
00053 T 0013
00054 T 0013
00055 T 0013
00056 T 0013
00057 T 0013
00058 T 0013
00059 T 0013
00060 T 0013
00061 T 0013
00062 T 0013
00063 T 0013
00064 T 0013
00065 T 0013
00066 T 0013
00067 T 0013
00068 T 0013
00069 T 0013
00070 T 0013
00071 T 0013
00072 T 0013
00073 T 0013
00074 T 0013
00075 T 0017
00076 T 0017
00077 T 0017
00078 T 0017
00079 T 0017
00080 T 0017
00081 T 0017
00082 T 0017
00083 T 0017
00084 T 0019
00085 T 0019
00086 T 0019
00087 T 0019
00088 T 0019
00089 T 0019
00090 T 0019
00091 T 0019
00092 T 0019
00093 T 0019
00094 T 0019
00095 T 0019
00096 T 0019
00097 T 0019
00098 T 0019
00099 T 0019
00100 T 0019
00101 T 0019
00102 T 0019
00103 T 0019
00104 T 0019
00105 T 0019

```

```

INTEGER NUMTYPES;%
%
%*** PARAMETER TABLE ***
ARRAY PARAMTAB[0:MAXPARAMS];%
DEFINE PARAMNAME=[9:10]#,%
        PARAMKIND=[13:14]#,%
        PARAMLEVEL=[23:10]#,%
        PARAMTYPE=[33:10]#,%
        PARAMFILE=[34:1]#;%
INTEGER NUMPARAMS;%
%
%*** CONSTANT TABLE ***
ARRAY CONSTTAB[0:MAXCONSTS];%
INTEGER NUMCONSTS;%
%
%*** LABEL TABLE ***
ARRAY LABTAB[0:MAXLABS];%
DEFINE LABVAL=[14:15]#,%
        LABDEF=[15:1]#;%
INTEGER NUMLABS,FIRSTLAB;%
%
%*** TABLES FOR I/O AND CHARACTER HANDLING ***
ARRAY CH[0:0],TEXT[0:1],STRING[0:11];%
        POINTER CHARPNT,TEXTPNT,TEXTPNT0,STRINGPNT;%
ARRAY ICARD[0:9],LINE[0:16],XLINE[0:10],ALGOLCARD[0:9];%
        POINTER CARDPNT,LINEPNT,XLINEPNT,ALGOLPNT;%
INTEGER CHARCNT,ALGOLCNT,MARGINCNT;%
ARRAY HEADTEXT[0:10],ERRLINE[0:16];%
INTEGER LINECNT,PAGECNT,ERRINX;%
%
%*** XREF FILE AND TABLE ***
FILE XREFFILE DISK SERIAL [20:3000] (2,3,150);%
ARRAY BLOCKTAB[0:MAXTABLES],XREFLINE[0:16];%
        INTEGER NUMXREF,NUMBLOCKS; POINTER XREFPNT;%
%
%*** OTHER TABLES ***
INTEGER ARRAY VARLIST[0:LISTLENGTH]; % TEMPORARY LIST OF VARIABLES,
        INTEGER VARINDEX,FIRSTVAR;%
ARRAY SYMTAB[0:MAXSYMS]; % USED BY "EXPRESSION",
        INTEGER NUMSYMS;%
ARRAY WITHTAB[0:MAXWITHSYMS]; % USED BY "WITHSTAT",
        INTEGER NWITHSYMS;%
INTEGER ARRAY SYMBOL[0:64]; % USED BY "INSYMBOL",
INTEGER ARRAY SYMKIND[0:61]; % USED IN ERROR RECOVERY,
ARRAY PNTRTAB1,PNTRTAB2,PNTRTAB3[0:MAXPNTRS];% USED FOR FORWARD POINTERS
        INTEGER NUMPNTRS;%
ARRAY EXTFILETAB[0:MAXEXTFILES]; % EXTERNAL FILES,
        INTEGER NUMEXTFILES;%
ARRAY FILETAB[0:MAXFILES]; % FILES IN USE,
        INTEGER NUMFILES;%
BOOLEAN ARRAY ERR[0:119]; % RECORDS ERROR MESSAGES,
%
%*** COMPILE TIME OPTIONS ***
BOOLEAN LISTOPTION,RESWORDOPTION,CHECKOPTION,DUMPOPTION,XREFOPTION;%
INTEGER CARDLNGTH;%
%
%*** INTRINSIC TYPES ***

```

```

00106 T 0019
00107 T 0019
00108 T 0019
00109 T 0019
00110 T 0021
00111 T 0021
00112 T 0021
00113 T 0021
00114 T 0021
00115 T 0021
00116 T 0021
00117 T 0021
00118 T 0021
00119 T 0023
00120 T 0023
00121 T 0023
00122 T 0023
00123 T 0025
00124 T 0025
00125 T 0025
00126 T 0025
00127 T 0025
00128 T 0025
00129 T 0030
00130 T 0030
00131 T 0037
00132 T 0037
00133 T 0037
00134 T 0040
00135 T 0040
00136 T 0040
00137 T 0040
00138 T 0044
00139 T 0047
00140 T 0047
00141 T 0047
00142 T 0047
00143 T 0049
00144 T 0049
00145 T 0051
00146 T 0051
00147 T 0053
00148 T 0053
00149 T 0054
00150 T 0056
00151 T 0058
00152 T 0058
00153 T 0060
00154 T 0060
00155 T 0062
00156 T 0062
00157 T 0064
00158 T 0064
00159 T 0064
00160 T 0064
00161 T 0064
00162 T 0064

```



```

INTEGER INTTYPE,REALTYPE,ALFATYPE,CHARTYPE,BOOLTYPE,NILTYPE,TEXTTYPE,%
  INPUTFILE,OUTPUTFILE,EMPTYSET;%
BOOLEAN INPUTDECL,OUTPUTDECL;%
%
%*** TEMPORARY VARIABLES ***
INTEGER T1,T2,T3,T4,T5;%
%
%*** OTHER VARIABLES ***
ALPHA USER; % THE USER NUMBER FOUND ON THE USER CARD.
%
INTEGER CURLEVEL; % CURRENT PROCEDURE LEVEL,
  TOPLEVEL; % TOP LEVEL IN DISPLAY VECTOR,
  NUMBEGINS; % NUMBER OF "BEGIN" S IN THE PROGRAM,
  NUMCASES; % NUMBER OF CASE-STATEMENTS IN PROGRAM,
  NUMREPS; % NUMBER OF REPEAT-STATEMENTS IN PROGRAM,
  NUMTEMPS; % NUMBER OF TEMPORARY VARIABLES IN USE,
  CURFUNC; % INDEX OF FUNCTION CURRENTLY COMPILED,
  CURSY; % LAST SYMBOL READ BY SCANNER,
  CURTYPE; % TYPE OF ENTITY LAST COMPILED,
  CURMODE; % CURRENT EXPRESSION MODE,
  LASTREC; % LAST RECORD TABLE DEFINED,
%
LABEL ENDOPINPUT;%
%
FORMAT NOERRORS ("NO ERRORS DETECTED,");%
%
  ERRORS (15," ERRORS DETECTED"/);%
  ALIST ("S SET LIST SINGLE");%
  NOALIST ("S RESET LIST");%
  LASTLINE (") TERMINATE: END OF PASCAL PROGRAM,");%
  TERMMESS ("**** END-OF-INPUT, COMPILATION TERMINATED,");%
%
MONITOR EXPOVR:=REALOVERFLOW;%
%
%*** SCANNER SYMBOLS ***
DEFINE IDENTIFIER=1#, INTCONST=2#, REALCONST=3#, ALFACONST=4#,%
  CHARCONST=5#, NOTSY=6#, ASTERISK=7#, SLASH=8#,%
  ANDSY=9#, DIVSY=10#, MODSY=11#, PLUS=12#,%
  MINUS=13#, ORSY=14#, LSSSY=15#, LEQSY=16#,%
  GEQSY=17#, GTRSY=18#, NEQSY=19#, EQLSY=20#,%
  INSY=21#, LPAR=22#, RPAR=23#, LBRACKET=24#,%
  RBRACKET=25#, DOUBLEDOT=26#, COMMA=27#, SEMICOLON=28#,%
  DGT=29#, ARROW=30#, COLON=31#, ASSIGNSY=32#,%
  BEGINSY=33#, ENDSY=34#, IFSY=35#, THENSY=36#,%
  ELSESY=37#, CASESY=38#, OFSY=39#, REPEATSY=40#,%
  UNTILSY=41#, WHILESY=42#, DOSY=43#, FORSY=44#,%
  TOSY=45#, DOWNTOSY=46#, GOTOSY=47#, NILSY=48#,%
  TYPESY=49#, ARRAYSY=50#, RECORDSY=51#, FILESY=52#,%
  SETSY=53#, CONSTSY=54#, VARSY=55#, LABELSY=56#,%
  FUNCSY=57#, PROCSY=58#, WITHSY=59#, PROGRAMSY=60#,%
  PACKEDSY=61#;%
%
DEFINE INITIAL=0#, MIDDLE=1#, TERMINAL=2#;%
DEFINE NUMBER=0#, BITPATTERN=1#;%

```

```

00163 T 0064
00164 T 0064
00165 T 0064
00166 T 0064
00167 T 0064
00168 T 0064
00169 T 0064
00170 T 0064
00171 T 0064
00172 T 0064
00173 T 0064
00174 T 0064
00175 T 0064
00176 T 0064
00177 T 0064
00178 T 0064
00179 T 0064
00180 T 0064
00181 T 0064
00182 T 0064
00183 T 0064
00184 T 0064
00185 T 0064
00186 T 0064
00187 T 0064
START OF SEGMENT ***** 3
00188 T 0064
00189 T 0064
00190 T 0064
00191 T 0064
00192 T 0064
3 IS 46 LONG, NEXT SEG 2
00193 T 0064
00194 T 0066
00195 T 0066
00196 T 0066
00197 T 0066
00198 T 0066
00199 T 0066
00200 T 0066
00201 T 0066
00202 T 0066
00203 T 0066
00204 T 0066
00205 T 0066
00206 T 0066
00207 T 0066
00208 T 0066
00209 T 0066
00210 T 0066
00211 T 0066
00212 T 0066
00213 T 0066
00214 T 0066

```

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
%
%
%
%
%
%
%
%
%
%
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
%
%
PROCEDURE INSYMBOL; FORWARD;%
PROCEDURE WRITEALGOL; FORWARD;%
PROCEDURE NEWXREF(NAME1,NAME2,TABLE,DECL);%
VALUE NAME1, NAME2, TABLE, DECL;%
REAL NAME1, NAME2;%
INTEGER TABLE;%
BOOLEAN DECL;%
FORWARD;%
%
DEFINE NDIGITS(N)=%
IF N≤ 9 THEN 1 ELSE%
IF N≤99 THEN 2 ELSE 3 DIGITS#;%
%
DEFINE HEADING=%
BEGIN COMMENT *** PRINTS A HEADING ON TOP OF A NEW PAGE. ;%
PAGECNT:=PAGECNT+1;%
REPLACE POINTER(HEADTEXT[*])+85 BY PAGECNT FOR NDIGITS(PAGECNT);%
WRITE(LINES[PAGE]);%
WRITE(LINES[DBL],11,HEADTEXT[*]);%
LINECNT:=2;%
END OF HEADING#;%
%
%
DEFINE PRINTLINE= %*** PRINTS A SOURCE CODE LINE.
BEGIN%
REPLACE LINEPNT=8 BY CARDNT FOR 5 DIGITS;%
IF LINECNT≥LINESPERPAGE THEN HEADING;%
IF RESWORDOPTION THEN%
BEGIN%
WRITE(LINES[NO],11,XLINE[*]);%
WRITE(LINES[NO],11,XLINE[*]);%
END;%
WRITE(LINES,17,LINE[*]);%
LINECNT:=LINECNT+1;%
END OF PRINTLINE#;%
%
%
DEFINE NEWCARD= %*** READS A NEW SOURCE CODE CARD.
BEGIN%
IF LISTOPTION THEN PRINTLINE;%
IF ERRINX>0 THEN PRINTERRORS;%
READ(CARD,10,ICARD[*]) [ENDOFINPUT];%
CARDPNT:=POINTER(ICARD[*]);%
REPLACE LINEPNT BY CARDPNT FOR 10 WORDS, " " FOR 6 WORDS;%
REPLACE XLINEPNT BY " " FOR 10 WORDS;%

```

```

00216 T 0066
00217 T 0066
00218 T 0066
00219 T 0068
00220 T 0066
00221 T 0066
00222 T 0066
00223 T 0066
00224 T 0066
00225 T 0066
00226 T 0066
00227 T 0066
00228 T 0066
00229 T 0069
00230 T 0069
00231 T 0069
00232 T 0069
00233 T 0069
00234 T 0069
00235 T 0069
00236 T 0069
00237 T 0069
00238 T 0069
00239 T 0069
00240 T 0069
00241 T 0069
00242 T 0069
00243 T 0069
00244 T 0069
00245 T 0069
00246 T 0069
00247 T 0069
00248 T 0069
00249 T 0069
00250 T 0069
00251 T 0069
00252 T 0069
00253 T 0069
00254 T 0069
00255 T 0069
00256 T 0069
00257 T 0069
00258 T 0069
00259 T 0069
00260 T 0069
00261 T 0069
00262 T 0069
00263 T 0069
00264 T 0069
00265 T 0069
00266 T 0069
00267 T 0069
00268 T 0069
00269 T 0069
00270 T 0069
00271 T 0069
00272 T 0069

```

```

CHARCNT:=CARDLENGTH;%
MARGINCNT:=85;%
CARDCNT:=CARDCNT+1;%
END#;%
%
%
DEFINE GEN(T,N,START)=      %*** GENERATE A TEXT "T", CONSISTING OF
BEGIN                      %*** "N" LETTERS, STARTING AT "START".
  IF ALGOLCNT<N THEN WRITEALGOL;%
  TEXT[0]:=T;%
  REPLACE ALGOLPNT:ALGOLPNT BY TEXTPNT0+START FOR N;%
  ALGOLCNT:=ALGOLCNT-N;%
END#;%
%
%
DEFINE GENID(L,NUM,NDIG)=   %*** GENERATE AN XALGOL IDENTIFIER.
BEGIN%
  IF ALGOLCNT<NDIG THEN WRITEALGOL;%
  CH[0]:=L;%
  REPLACE ALGOLPNT:ALGOLPNT BY CHARPNT FOR 1, NUM FOR NDIG DIGITS;%
  ALGOLCNT:=ALGOLCNT-NDIG-1;%
END#;%
%
%
INTEGER NABS,NSIZE;%
%
DEFINE GENINT(N)=%
BEGIN%
  NABS:=ABS(N); IF N<0 THEN GEN("=",1,7);%
  NSIZE:=IF NABS 9 THEN 1 ELSE%
          IF NABS 999 THEN 3 ELSE%
          IF NABS 99999 THEN 5 ELSE%
          IF NABS<999999999 THEN 8 ELSE 12;%
  IF ALGOLCNT<NSIZE THEN WRITEALGOL;%
  IF NSIZE=12 THEN%
  REPLACE ALGOLPNT:ALGOLPNT BY (NABS DIV 1000000) FOR 6 DIGITS,%
  ENTIER(NABS MOD 1000000) FOR 6 DIGITS ELSE%
  REPLACE ALGOLPNT:ALGOLPNT BY NABS FOR NSIZE DIGITS;%
  ALGOLCNT:=ALGOLCNT-NSIZE;%
END OF GENINT#;%
%
%
PROCEDURE GENREAL(X);%
VALUE X; REAL X;%
BEGIN%
  REAL ABSX;%
  INTEGER POWER,V1,V2;%
  %
  IF X.[46:5]=0 THEN%
  BEGIN%
    IF ALGOLCNT<9 THEN WRITEALGOL;%
    TEXT[0]:=X;%
    REPLACE ALGOLPNT:ALGOLPNT BY "", TEXTPNT FOR 7, "";%
    ALGOLCNT:=ALGOLCNT-9;%
  END ELSE%

```

```

00273 T 0069
00274 T 0069
00275 T 0069
00276 T 0069
00277 T 0069
00278 T 0069
00279 T 0069
00280 T 0069
00281 T 0069
00282 T 0069
00283 T 0069
00284 T 0069
00285 T 0069
00286 T 0069
00287 T 0069
00288 T 0069
00289 T 0069
00290 T 0069
00291 T 0069
00292 T 0069
00293 T 0069
00294 T 0069
00295 T 0069
00296 T 0069
00297 T 0069
00298 T 0069
00299 T 0069
00300 T 0069
00301 T 0069
00302 T 0069
00303 T 0069
00304 T 0069
00305 T 0069
00306 T 0069
00307 T 0069
00308 T 0069
00309 T 0069
00310 T 0069
00311 T 0069
00312 T 0069
00313 T 0069
00314 T 0069
00315 T 0069
00316 T 0069
00317 T 0069
00318 T 0069
00319 T 0000
00320 T 0000
00321 T 0000
00322 T 0001
00323 T 0001
00324 T 0003
00325 T 0004
00326 T 0013
00327 T 0015

```

START OF SEGMENT ***** 4

```

BEGIN%
IF ALGOLCNT<22 THEN WRITEALGOL;%
IF X<0 THEN GEN("=",2,6);%

ABSX:=ABS(X);%
IF ABSX>0 THEN%
BEGIN%
  WHILE ABSX≥1@7 DO BEGIN ABSX:=ABSX/10; POWER:=POWER+1 END;%

  WHILE ABSX<1@6 DO BEGIN ABSX:=ABSX*10; POWER:=POWER-1 END;%

  V1:=ENTIER(ABSX);%
  V2:=ENTIER((ABSX-V1)*1000000);%
  REPLACE ALGOLPNT:ALGOLPNT BY V1 FOR 7 DIGITS, ".";%
  V2 FOR 6 DIGITS, "@";%
  ALGOLCNT:=ALGOLCNT-15;%
  IF POWER<0 THEN GEN("=",1,7);%

  POWER:=ABS(POWER);%
  REPLACE ALGOLPNT:ALGOLPNT BY POWER FOR 2 DIGITS;%
  ALGOLCNT:=ALGOLCNT-2;%
  END ELSE GEN("0",1,7);%

  IF X<0 THEN GEN(")",1,7);%
END;%
END OF GENREAL;%

```

```

00328 T 0015
00329 T 0015
00330 T 0017
00000330
00331 T 0027
00332 T 0028
00333 T 0029
00334 T 0029
00000334
00335 T 0037
00000335
00336 T 0043
00337 T 0044
00338 T 0047
00339 T 0055
00340 T 0061
00341 T 0062
00000341
00342 T 0073
00343 T 0074
00344 T 0078
00345 T 0079
00000333
00000345
00346 T 0091
00000346
00347 T 0101
00000328
00348 T 0101
00000317
4 IS 105 LONG, NEXT SEG 2

```

```

%
%
INTEGER TYPEINDEX;%
%
DEFINE NEWTYPE=%
BEGIN%
  IF NUMTYPES≥MAXTYPES THEN BEGIN ERROR(45);NUMTYPES:=MAXTYPES-20 END;
  TYPEINDEX:=NUMTYPES:=NUMTYPES+1;%
END #;%
%
%
PROCEDURE WRITEALGOL;      %*** WRITES A COMPLETED XALGOL CARD TO
BEGIN                      %*** THE FILE.
  REPLACE POINTER(ALGOLCARD[9]) BY CARDNT FOR 8 DIGITS;%
  WRITE(PASCALGOL,10,ALGOLCARD[*]);%
  IF DUMPTION THEN WRITE(LINES,10,ALGOLCARD[*]);%
  ALGOLPNT:=POINTER(ALGOLCARD[*]); ALGOLCNT:=71;%
  REPLACE ALGOLPNT BY " " FOR 9 WORDS;%
END OF WRITEALGOL;%

```

```

00349 T 0069
00350 T 0069
00351 T 0069
00352 T 0069
00353 T 0069
00354 T 0069
00355 T 0069
00356 T 0069
00357 T 0069
00358 T 0069
00359 T 0069
00360 T 0069
00361 T 0069
00362 T 0069
00363 T 0075
00364 T 0079
00365 T 0084
00366 T 0086
00367 T 0090
00000361

```

```

%
%
DEFINE MARGIN(LETTER,NUM)=%
BEGIN COMMENT *** PLACES INFORMATION IN THE MARGIN. ;%
  IF MARGINCNT<=118 THEN%
  BEGIN TEXT[0]=LETTER;%
    REPLACE LINEPNT+MARGINCNT BY TEXTPNT+5 FOR 2,%
      NUM FOR NDIGITS(NUM));%
    MARGINCNT:=MARGINCNT+6;%
  END;%
END OF MARGIN#;%
%
%
PROCEDURE SKIP(SYMBOL);      *** SKIP SYMBOLS TO RECOVER FROM ERROR
VALUE SYMBOL; INTEGER SYMBOL; *** CONDITION.
BEGIN%
  WHILE CURSY#SYMBOL AND SYMKIND[CURSY]=MIDDLE DO%
  IF CURSY=RECORDSY THEN%
  BEGIN DO BEGIN INSYMBOL;%
    SKIP(99);%
    END UNTIL CURSY#SEMICOLON AND CURSY#CASESY;%
  END ELSE INSYMBOL;%
END OF SKIP;%

```

```

00368 T 0091
00369 T 0091
00370 T 0091
00371 T 0091
00372 T 0091
00373 T 0091
00374 T 0091
00375 T 0091
00376 T 0091
00377 T 0091
00378 T 0091
00379 T 0091
00380 T 0091
00381 T 0091
00382 T 0091
00383 T 0091
00384 T 0091
00385 T 0091
00386 T 0094
00387 T 0095
00388 T 0096
00389 T 0098 00000386
00390 T 0100 00000386
00000388

```

```

%
%
PROCEDURE ERROR(ERRNUM);%
VALUE ERRNUM; INTEGER ERRNUM;%
BEGIN COMMENT *** ARRANGE ERROR INDICATOR. ;%
  NUMERRS:=NUMERRS+1;%
  ERR[ERRNUM]:=TRUE;%
  ERRINX:=MAX(ERRINX,CARDLENGTH-2-CHARCNT);%
  IF ERRINX<=115 THEN%
  BEGIN REPLACE POINTER(ERRLINE[1])+ERRINX BY "X",%
    ERRNUM FOR NDIGITS(ERRNUM));%
    ERRINX:=ERRINX+(IF ERRNUM<=9 THEN 2 ELSE%
      IF ERRNUM<=99 THEN 3 ELSE 4));%
  END END OF ERROR;%

```

```

00391 T 0100
00392 T 0100
00393 T 0100
00394 T 0100
00395 T 0100
00396 T 0100
00397 T 0102
00398 T 0103
00399 T 0107
00400 T 0108
00401 T 0116
00402 T 0123
00403 T 0126
00404 T 0129
00000400
00000395

```

```

%
%
PROCEDURE PRINTERRORS;%
BEGIN COMMENT *** PRINT ERROR INDICATORS. ;%
  IF NOT LISTOPTION THEN PRINTLINE;%

```

```

00405 T 0129
00406 T 0129
00407 T 0129
00408 T 0129
00409 T 0129
00000409
00000409

```

```

WRITE(LINES,17,ERRLINE[*]);%
LINECNT:=LINECNT+1;%
REPLACE POINTER(ERRLINE[1]) BY " " FOR 16 WORDS;%
ERRINX:=0;%
END OF PRINTERRORS;%

```

```

00000409
00410 T 0174
00411 T 0178
00412 T 0179
00413 T 0185
00414 T 0186
00000408

```

```

%
%
DEFINE HASH(N) = (N),[35:36] MOD MAXNAMES#;%
%
INTEGER THISLEVEL,THISTAB,THISINDEX;%
ALPHA THISID,TNAME;%
BOOLEAN FOUND;%
%
DEFINE SEARCHTAB(TAB)=          %*** SEARCH NAME TABLE "TAB" FOR THE
BEGIN                          %*** IDENTIFIER JUST READ.
    THISINDEX:=HASH(CURNAME1);%
    TNAME:=NAMETAB1[TAB,THISINDEX];%
    WHILE (IF TNAME=CURNAME1 THEN NAMETAB2[TAB,THISINDEX]#CURNAME2%
           ELSE TNAME#0) DO%
        BEGIN%
            THISINDEX:=IF THISINDEX=0 THEN MAXNAMES ELSE THISINDEX-1;%
            TNAME:=NAMETAB1[TAB,THISINDEX];%
        END;%
        FOUND:=TNAME#0;%
        IF XREFOPTION THEN%
            IF FOUND THEN NEWXREF(CURNAME1,CURNAME2,TAB,FALSE); %
END OF SEARCHTAB#;%
%
DEFINE SEARCH=                  %*** SEARCH ALL TABLES CURRENTLY IN USE.
BEGIN%
    THISLEVEL:=TOPLEVEL+1;%
    DO BEGIN%
        THISLEVEL:=THISLEVEL-1;%
        THISTAB:=IF THISLEVEL#CURLEVEL THEN THISLEVEL%
                 ELSE DISPLAY[THISLEVEL],NAMETAB;%
        SEARCHTAB(THISTAB);%
    END UNTIL FOUND OR THISLEVEL=0;%
    THISID:=NAMETAB3[THISTAB,THISINDEX];%
END OF SEARCH #;%
%
%
DEFINE NEWNAME(NAME1,NAME2,TAB) =%
BEGIN                          %*** ENTER A NEW NAME INTO THE NAME TABLE "TAB".
    THISINDEX:=HASH(NAME1);%
    TNAME:=NAMETAB1[TAB,THISINDEX];%
    WHILE(IF TNAME=NAME1 THEN NAMETAB2[TAB,THISINDEX]#NAME2%
           ELSE TNAME#0) DO%
        BEGIN%
            THISINDEX:=IF THISINDEX=0 THEN MAXNAMES ELSE THISINDEX-1;%
            TNAME:=NAMETAB1[TAB,THISINDEX];%
        END;%

```

```

00415 T 0186
00416 T 0186
00417 T 0186
00418 T 0186
00419 T 0186
00420 T 0186
00421 T 0186
00422 T 0186
00423 T 0186
00424 T 0186
00425 T 0186
00426 T 0186
00427 T 0186
00428 T 0186
00429 T 0186
00430 T 0186
00431 T 0186
00432 T 0186
00433 T 0186
00434 T 0186
00435 T 0186
00436 T 0186
00437 T 0186
00438 T 0186
00439 T 0186
00440 T 0186
00441 T 0186
00442 T 0186
00443 T 0186
00444 T 0186
00445 T 0186
00446 T 0186
00447 T 0186
00448 T 0186
00449 T 0186
00450 T 0186
00451 T 0186
00452 T 0186
00453 T 0186
00454 T 0186
00455 T 0186
00456 T 0186
00457 T 0186
00458 T 0186
00459 T 0186
00460 T 0186

```

```

IF TNAME#0 THEN ERROR(2);%
NAMETAB1[TAB,THISINDEX]:=NAME1;%
NAMETAB2[TAB,THISINDEX]:=NAME2;%
IF XREFOPTION THEN NEWXREF(NAME1,NAME2,TAB,TRUE);%
END OF NEWNAME #;%
%
%
PROCEDURE INITIALIZE;
BEGIN
INTEGER T1,T3;%

ALPHA A;%
FILL SYMKIND[*] WITH 28(MIDDLE),TERMINAL,4(MIDDLE),INITIAL,TERMINAL,
INITIAL,MIDDLE,TERMINAL,INITIAL,MIDDLE,INITIAL,TERMINAL,INITIAL,%
MIDDLE,INITIAL,2(MIDDLE),INITIAL,MIDDLE,INITIAL,4(MIDDLE),%
7(INITIAL),MIDDLE;%

%
FILL SYMBOL[*] WITH 10(0),0,ARROW,0,COLON,GTRSY,GEQSY,PLUS,9(0),%
DOT,LBRACKET,ANDSY,LPAR,LSSSY,ARROW,0,9(0),0,ASTERISK,MINUS,%
RPAR,SEMICOLON,LEQSY,0,SLASH,8(0),COMMA,0,NEQSY,EQLSY,RBRACKET,%
0,DOUBLEDOT;%

%
LINEPNT :=POINTER(LINE[1]);%
XLINEPNT:=POINTER(XLINE[1]);%
REPLACE LINEPNT=8 BY " " => " " FOR 16 WORDS;%
REPLACE XLINEPNT=8 BY " " FOR 11 WORDS;%
REPLACE POINTER(ERRLINE[*]) BY "**** " FOR 16 WORDS;%
ALGOLPNT:=POINTER(ALGOLCARD[*]); ALGOLCNT:=71;%
REPLACE ALGOLPNT BY " " FOR 9 WORDS;%
CHARPNT:=POINTER(CH[*])+7;%
TEXTPNT:=POINTER(TEXT[*])+1; TEXTPNT0:=TEXTPNT-1;%
REPLACE TEXTPNT BY " " FOR 15;%
STRINGPNT:=POINTER(STRING[*]);%
REPLACE POINTER(HEADTEXT[*]) BY " " FOR 10 WORDS, "PAGE ";%
REPLACE POINTER(HEADTEXT[*]) BY "PASCAL(", EDITION, ")/B=5700";%
TEXT[0]:=TIME(5);%
REPLACE POINTER(HEADTEXT[*])+45 BY TEXTPNT+3 FOR 2, "/" ,%
TEXTPNT+1 FOR 2, " ", TEXTPNT+5 FOR 2;%
T1:=TIME(1)/3600;%
REPLACE POINTER(HEADTEXT[*])+57 BY (T1 DIV 60) FOR 2 DIGITS, ":",%
ENTIER(T1 MOD 60) FOR 2 DIGITS;%
HEADING;%

%
%
%*** INITIALIZE INTRINSIC TYPES, CONSTANTS ETC. ***
%
INTTYPE:=T3:=1; %*** "INTEGER" ***
T1:=NUMERIC; T1,SIZE:=1; T1,STRUCT:=0;%
TYPETAB1[1]:=T1; TYPETAB2[1]:=MAXINT; TYPETAB3[1]:=MAXINT;%
NEWNAME("7INTEGE", "R", 0); T3.IDCLASS:=TYPES;%

NAMETAB3[0,THISINDEX]:=T3;%

```

```

00461 T 0186
00462 T 0186
00463 T 0186
00464 T 0186
00465 T 0186
00466 T 0186
00467 T 0186
00468 T 0186
00469 T 0186
00470 T 0186
START OF SEGMENT ***** 5
00471 T 0000
00472 T 0000
START OF SEGMENT ***** 6
00473 T 0001
00474 T 0001
00475 T 0001
6 IS 62 LONG, NEXT SEG 5
00476 T 0001
00477 T 0001
START OF SEGMENT ***** 7
00478 T 0003
00479 T 0003
00480 T 0003
7 IS 65 LONG, NEXT SEG 5
00481 T 0003
00482 T 0003
00483 T 0006
00484 T 0009
00485 T 0018
00486 T 0024
00487 T 0033
00488 T 0036
00489 T 0040
00490 T 0043
00491 T 0049
00492 T 0053
00493 T 0055
00494 T 0064
00495 T 0075
00496 T 0076
00497 T 0087
00498 T 0098
00499 T 0100
00500 T 0111
00501 T 0113
00000501
00502 T 0136
00503 T 0136
00504 T 0136
00505 T 0136
00506 T 0138
00507 T 0142
00508 T 0146
00000508
00000508
00509 T 0177

```

REALTYPE:=T3:=2; %*** "REAL" ***
T1.FORM:=FLOATING; TYPETAB1[2]:=T1;%
NEWNAME("400REAL",0,0); T3.IDCLASS:=TYPES;%

00510 T 0179
00511 T 0180
00512 T 0183
00000512
00000512

NAMETAB3[0,THISINDEX]:=T3;%
ALFATYPE:=T3:=3; %*** "ALFA" ***
T1.FORM:=ALFA; TYPETAB1[3]:=T1;%
NEWNAME("400ALFA",0,0); T3.IDCLASS:=TYPES;%

00513 T 0210
00514 T 0212
00515 T 0213
00516 T 0216
00000516
00000516

NAMETAB3[0,THISINDEX]:=T3;%
BOOQTYPE:=T3:=4; %*** "BOOLEAN" ***
T1.FORM:=SYMBOLIC; TYPETAB1[4]:=T1; TYPETAB3[4]:=1;%
NEWNAME("7BOOLEA","N",0); T3.IDCLASS:=TYPES;%

00517 T 0243
00518 T 0245
00519 T 0246
00520 T 0250
00000520
00000520

NAMETAB3[0,THISINDEX]:=T3;%
CHARTYPE:=T3:=5; %*** "CHAR" ***
T1.FORM:=CHAR; TYPETAB1[5]:=T1; TYPETAB3[5]:=63;%
NEWNAME("400CHAR",0,0); T3.IDCLASS:=TYPES;%

00521 T 0277
00522 T 0279
00523 T 0280
00524 T 0284
00000524
00000524

NAMETAB3[0,THISINDEX]:=T3;%
T3:=BOOQTYPE; T3.IDCLASS:=CONST; %*** "FALSE" ***
NEWNAME("50FALSE",0,0); NAMETAB3[0,THISINDEX]:=T3;%

00525 T 0311
00526 T 0313
00527 T 0315
00000527
00000527

T3.INFO:=1; %*** "TRUE" ***
NEWNAME("400TRUE",0,0); NAMETAB3[0,THISINDEX]:=T3;%

00528 T 0342
00529 T 0344
00000529
00000529

NUMTYPES:=5;%
NILTYPE:=1; %*** TYPE OF "NIL" ***
EMPTYSET:=2; %*** TYPE OF [] ***
NEWNAME("6MAXINT",0,0); T3:=INTTYPE; %*** "MAXINT" ***

00530 T 0371
00531 T 0372
00532 T 0373
00533 T 0374
00000533
00000533

T3.IDCLASS:=CONST; T3.INFO:=1024;%
NAMETAB3[0,THISINDEX]:=T3;%
NUMCONSTS:=1; CONSTTAB[1]:=MAXINT;%

00534 T 0400
00535 T 0403
00536 T 0405
00537 T 0407
00538 T 0407
00539 T 0410
00540 T 0420
00541 T 0430
00542 T 0430

T3:=0; T3.IDCLASS:=PROC; %*** PROCEDURES ***
FOR A:="300GET", "300NEW", "400PACK", "400PAGE", "3000PUT",
"400READ", "6READLN", "50RESET", "6UNPACK", "50WRITE" DO
BEGIN%
NEWNAME(A,0,0); NAMETAB3[0,THISINDEX]:=T3;%

00000542
00000542

END;%

00543 T 0468
00000541

NEWNAME("7DISPOS","E",0); NAMETAB3[0,THISINDEX]:=T3;%

00544 T 0468
00000544
00000544

NEWNAME("7REWRIT","E",0); NAMETAB3[0,THISINDEX]:=T3;%

00545 T 0495
00000545
00000545


```

PROCEDURE NEWXREF(NAME1,NAME2,TABLE,DECL);%
VALUE NAME1,NAME2,TABLE,DECL;%
REAL NAME1,NAME2;%
INTEGER TABLE;%
BOOLEAN DECL;%
BEGIN%
  NL:=NAME1.NAMELENGTH;%
  IF NL<7 THEN NAME1:=0&NAME1[41:41:6]&NAME1[35:6*NL-1:6*NL];%
  ELSE NAME2:=0&NAME2[35:6*(NL-6)-1:6*(NL-6)];%
  AX:=CARDCNT; AX.XREFBLOCK:=BLOCKTAB[TABLE];%
  IF DECL THEN AX:=AX+100000000000;%
  WRITE(XREFFILE,*,NAME1,NAME2,AX);%
END OF NEWXREF;%

```

```

00582 T 0186
00583 T 0186
00584 T 0186
00585 T 0186
00586 T 0186
00587 T 0186
00588 T 0186
00589 T 0186
00590 T 0195
00591 T 0202
00592 T 0204
00593 T 0206
00594 T 0218

```

00000587

```

%
PROCEDURE XREFMAX(A);%
ARRAY A[0];%
BEGIN%
  A[0]:="AZZZZZ"; A[1]:="ZZZZZZ"; A[2]:="9999999999";%
END OF XREFMAX;%

```

```

00595 T 0218
00596 T 0218
00597 T 0218
00598 T 0218
00599 T 0218
00600 T 0222

```

00000598

```

%
%
BOOLEAN PROCEDURE XREFCOMPARE(A,B);%
ARRAY A,B[0];%
BEGIN%
  A0:=A[0]; B0:=B[0]; A1:=A[1]; B1:=B[1];%
  XREFCOMPARE:=%
  IF A0.[35:36]≠B0.[35:36] THEN A0.[35:36]<B0.[35:36] ELSE%
  IF A1≠B1 THEN A1<B1 ELSE%
  IF A0≠B0 THEN A0.NAMELENGTH<B0.NAMELENGTH ELSE%
  A[2] LEQ B[2];%
END OF XREFCOMPARE;%

```

```

00601 T 0226
00602 T 0226
00603 T 0226
00604 T 0226
00605 T 0226
00606 T 0226
00607 T 0230
00608 T 0230
00609 T 0234
00610 T 0237
00611 T 0240
00612 T 0242

```

00000605

```

%
%
PROCEDURE PRINTXREF(FINIS,A);%
VALUE FINIS; BOOLEAN FINIS;%
ARRAY A[0];%
BEGIN%
  IF FINIS THEN%
  BEGIN%
    WRITE(LINES,17,XREFLINE[*]);%
    CLOSE(LINES);%
    CLOSE(XREFFILE);%
  END%

```

```

00613 T 0245
00614 T 0245
00615 T 0245
00616 T 0245
00617 T 0245
00618 T 0245
00619 T 0245
00620 T 0245
00621 T 0245
00622 T 0250
00623 T 0251

```

```

END%
ELSE%
BEGIN%
AO:=A[0]; A1:=A[1]; A2:=A[2];%
IF AO=LASTAO AND A1=LASTA1 AND A2,XREFBLOCK=LASTBLOCK THEN%
BEGIN%
IF NUMXREF=15 THEN%
BEGIN%
WRITE(LINES,17,XREFLINE[*]); LINECNT:=LINECNT+1;%
IF LINECNT>LINESPERPAGE THEN HEADING;%

XREFPNT:=POINTER(XREFLINE[*]); NUMXREF:=0;%
REPLACE XREFPNT BY " " FOR 17 WORDS; XREFPNT:=XREFPNT+24;%
END;%

REPLACE XREFPNT BY A2,XREFCARD FOR 5 DIGITS;%
XREFPNT:=XREFPNT+7; NUMXREF:=NUMXREF+1;%
END ELSE%

IF A2<0 THEN%
BEGIN%
A2:=A2+1000000000000;%
WRITE(LINES,17,XREFLINE[*]); LINECNT:=LINECNT+1;%
IF LINECNT>LINESPERPAGE THEN HEADING;%

XREFPNT:=POINTER(XREFLINE[*]); NUMXREF:=0;%
REPLACE XREFPNT BY " " FOR 17 WORDS;%
TEXT[0]:=AO,[35:36]; LASTAO:=AO;%
REPLACE XREFPNT BY TEXTPNT+1 FOR AO,NAMELENGTH;%
TEXT[0]:=LASTA1:=A1;%
IF AO,NAMELENGTH>6 THEN%
REPLACE XREFPNT+6 BY TEXTPNT+1 FOR AO,NAMELENGTH-6;%
REPLACE XREFPNT+17 BY A2,XREFCARD FOR 5 DIGITS;%
XREFPNT:=XREFPNT+24; LASTBLOCK:=A2,XREFBLOCK;%
END;%

END;%
END OF PRINTXREF;%

```

```

%
%
%
INTEGER TT1,TT2,F1,F2,LT,RT;%
%
DEFINE CHECKTYPES(LEFTTYPE,RIGHTTYPE)=%
BEGIN%
IF LEFTTYPE>0 AND RIGHTTYPE>0 THEN%
IF LEFTTYPE≠RIGHTTYPE THEN%
BEGIN%
LT:=LEFTTYPE; RT:=RIGHTTYPE;%
TT1:=TYPETAB1[LT]; TT2:=TYPETAB1[RT];%

```

```

00624 T 0253 00000620
00625 T 0253
00626 T 0253
00627 T 0254
00628 T 0257
00629 T 0260
00630 T 0260
00631 T 0261
00632 T 0262
00633 T 0267 00000633
00634 T 0289
00635 T 0292
00636 T 0298 00000631
00637 T 0298
00638 T 0303
00639 T 0307 00000629
00640 T 0307
00641 T 0308
00642 T 0309
00643 T 0310
00644 T 0315 00000644
00645 T 0338
00646 T 0340
00647 T 0344
00648 T 0346
00649 T 0352
00650 T 0353
00651 T 0355
00652 T 0363
00653 T 0369
00654 T 0373 00000641
00655 T 0373 00000626
00656 T 0373 00000618
00657 T 0375
00658 T 0375
00659 T 0375
00660 T 0375
00661 T 0375
00662 T 0375
00663 T 0375
00664 T 0375
00665 T 0375
00666 T 0375
00667 T 0375
00668 T 0375

```

```

F1:=TT1,FORM;      F2:=TT2,FORM;%
IF L#REALTYPE OR F2#NUMERIC THEN%
IF(F1#SET AND LT#EMPTYSET)OR(F2#SET AND RT#EMPTYSET)THEN%
IF(F1#POINTERS AND LT#NILTYPE)OR(F2#POINTERS AND RT#NILTYPE)THEN%
BEGIN%
  IF F1=SET AND F2=SET THEN%
  BEGIN%
    LT:=TT1,SETTYPE;      RT:=TT2,SETTYPE;%
    TT1:=TYPETAB1[LT];    TT2:=TYPETAB1[RT];%
    F1:=TT1,FORM;        F2:=TT2,FORM;%
  END;%
  IF F1=POINTERS AND F2=POINTERS THEN%
  BEGIN%
    LT:=TT1,POINTTYPE;    RT:=TT2,POINTTYPE;%
    TT1:=TYPETAB1[LT];    TT2:=TYPETAB1[RT];%
    F1:=TT1,FORM;        F2:=TT2,FORM;%
  END;%
  WHILE F1=SUBTYPE DO%
  BEGIN LT:=TT1,MAINTYPE; TT1:=TYPETAB1[LT]; F1:=TT1,FORM END;%
  WHILE F2=SUBTYPE DO%
  BEGIN RT:=TT2,MAINTYPE; TT2:=TYPETAB1[RT]; F2:=TT2,FORM END;%
  IF LT>0 AND RT>0 THEN%
  IF LT#RT THEN%
  IF F1#NUMERIC OR F2#NUMERIC THEN%
  IF F1#CHAR OR F2#CHAR THEN ERROR(17);%
  END;%
END;%
END OF CHECKTYPES#;%
%
%
INTEGER FILENAME;%
BOOLEAN LPARFOUND;%
%
DEFINE FILEPARAM(DEFAULTFILE)=%*** CHECKS THE FIRST PARAMETER TO SEE
BEGIN                                     %*** IF IT IS A FILE,
  INSYMBOL; FILENAME:=CURTYPE:=0;%
  LPARFOUND:=CURSY=LPAR;%
  IF LPARFOUND THEN%
  BEGIN%
    INSYMBOL;%
    IF CURSY=IDENTIFIER THEN%
    BEGIN%
      SEARCH;%
      IF FOUND THEN%
      BEGIN%
        IF THISID.IDCLASS=VAR THEN%
        BEGIN%
          CURTYPE:=THISID,TYPE;%
          IF TYPETAB1[CURTYPE].FORM#FILES THEN%
          BEGIN%
            FILENAME:=1000*THISLEVEL+THISINDEX;%
            INSYMBOL;%
          END END END END;%
          IF SYMKIND[CURSY]=TERMINAL THEN ERROR(46);%
        END;%
      IF FILENAME=0 THEN FILENAME:=DEFAULTFILE;%
      IF (FILENAME=INPUTFILE AND NOT INPUTDECL) OR%

```

```

00669 T 0375
00670 T 0375
00671 T 0375
00672 T 0375
00673 T 0375
00674 T 0375
00675 T 0375
00676 T 0375
00677 T 0375
00678 T 0375
00679 T 0375
00680 T 0375
00681 T 0375
00682 T 0375
00683 T 0375
00684 T 0375
00685 T 0375
00686 T 0375
00687 T 0375
00688 T 0375
00689 T 0375
00690 T 0375
00691 T 0375
00692 T 0375
00693 T 0375
00694 T 0375
00695 T 0375
00696 T 0375
00697 T 0375
00698 T 0375
00699 T 0375
00700 T 0375
00701 T 0375
00702 T 0375
00703 T 0375
00704 T 0375
00705 T 0375
00706 T 0375
00707 T 0375
00708 T 0375
00709 T 0375
00710 T 0375
00711 T 0375
00712 T 0375
00713 T 0375
00714 T 0375
00715 T 0375
00716 T 0375
00717 T 0375
00718 T 0375
00719 T 0375
00720 T 0375
00721 T 0375
00722 T 0375
00723 T 0375
00724 T 0375
00725 T 0375

```

```

        (FILENAME=OUTPUTFILE AND NOT OUTPUTDECL) THEN ERROR(96));%
END OF FILEPARAM#;%
%
%
INTEGER TFORM;%
BOOLEAN SIGNED,NEGATIVE;%
%
DEFINE CONSTANT(CVAL,CTYPE)=                %*** <CONSTANT> ***
BEGIN                                        %*****
  IF CURSY=MINUS OR CURSY=PLUS THEN%
  BEGIN SIGNED:=TRUE; NEGATIVE:=CURSY=MINUS;%
    INSYMBOL;%
  END ELSE SIGNED:=NEGATIVE:=FALSE;%
  IF CURSY=INTCONST THEN%
  BEGIN CTYPE:=INTTYPE;%
    CVAL:=IF NEGATIVE THEN -CURVAL ELSE CURVAL;%
  END ELSE%
  IF CURSY=CHARCONST THEN%
  BEGIN IF SIGNED THEN ERROR(29));%
    CTYPE:=CHARTYPE; CVAL:=CURVAL;%
  END ELSE%
  IF CURSY=REALCONST THEN%
  BEGIN CTYPE:=REALTYPE;%
    CVAL:=IF NEGATIVE THEN -CURVAL ELSE CURVAL;%
  END ELSE%
  IF CURSY=ALFACONST THEN%
  BEGIN IF SIGNED THEN ERROR(29));%
    IF CURLNGTH>7 THEN ERROR(41));%
    CTYPE:=ALFATYPE; CVAL:=CURVAL;%
  END ELSE%
  IF CURSY=IDENTIFIER THEN%
  BEGIN%
    SEARCH;%
    IF FOUND THEN%
    BEGIN%
      IF THISID.IDCLASS=CONST AND NOT BOOLEAN(THISID,FORMAL) THEN%
      BEGIN%
        IF TYPETAB1[THISID.TYPE].FORMSALFA THEN%
        BEGIN%
          CVAL:=THISID.INFO;%
          IF CVAL>1023 THEN CVAL:=CONSTTAB[CVAL-1023];%
          CTYPE:=THISID.TYPE;%
          IF SIGNED THEN%
          BEGIN%
            TFORM:=TYPETAB1[THISID.TYPE].FORM;%
            IF TFORM#NUMERIC AND TFORM#FLOATING THEN ERROR(29) ELSE%
            IF NEGATIVE THEN CVAL:=-CVAL;%
          END;%
        END ELSE BEGIN ERROR(48); CVAL:=CTYPE:=0 END;%
        END ELSE BEGIN ERROR(32); CVAL:=CTYPE:=0 END;%
        END ELSE BEGIN ERROR(1); CVAL:=CTYPE:=0 END;%
        END ELSE BEGIN ERROR(32); CVAL:=CTYPE:=0 END;%
        INSYMBOL;%
      END OF CONSTANT#;%

```

```

00726 T 0375
00727 T 0375
00728 T 0375
00729 T 0375
00730 T 0375
00731 T 0375
00732 T 0375
00733 T 0375
00734 T 0375
00735 T 0375
00736 T 0375
00737 T 0375
00738 T 0375
00739 T 0375
00740 T 0375
00741 T 0375
00742 T 0375
00743 T 0375
00744 T 0375
00745 T 0375
00746 T 0375
00747 T 0375
00748 T 0375
00749 T 0375
00750 T 0375
00751 T 0375
00752 T 0375
00753 T 0375
00754 T 0375
00755 T 0375
00756 T 0375
00757 T 0375
00758 T 0375
00759 T 0375
00760 T 0375
00761 T 0375
00762 T 0375
00763 T 0375
00764 T 0375
00765 T 0375
00766 T 0375
00767 T 0375
00768 T 0375
00769 T 0375
00770 T 0375
00771 T 0375
00772 T 0375
00773 T 0375
00774 T 0375
00775 T 0375
00776 T 0375
00777 T 0375
00778 T 0375
00779 T 0375

```

PART 31 THE SCANNER,

SYMBOL	INTERNAL NUMBER	INTERNAL NAME	SYMBOL KIND
IDENTIFIER	1	IDENTIFIER	MIDDLE
122	2	INTCONST	MIDDLE
2.5	3	REALCONST	MIDDLE
"ABCD"	4	ALFACONST	MIDDLE
"C"	5	CHARCONST	MIDDLE
NOT	6	NOTSY	MIDDLE
*	7	ASTERISK	MIDDLE
/	8	SLASH	MIDDLE
& AND	9	ANDSY	MIDDLE
DIV	10	DIVSY	MIDDLE
MOD	11	MODSY	MIDDLE
+	12	PLUS	MIDDLE
-	13	MINUS	MIDDLE
OR	14	ORSY	MIDDLE
< LSS	15	LSSSY	MIDDLE
<# LEQ	16	LEQSY	MIDDLE
># GEQ	17	GEQSY	MIDDLE
> GTR	18	GTRSY	MIDDLE
<> NEQ	19	NEQSY	MIDDLE
# EQL	20	EQLSY	MIDDLE
IN	21	INSY	MIDDLE
(22	LPAR	MIDDLE
)	23	RPAR	MIDDLE
[24	LBRACKET	MIDDLE
]	25	RBRACKET	MIDDLE
.	26	DOUBLEDOT	MIDDLE
,	27	COMMA	MIDDLE
;	28	SEMICOLON	TERMINAL
.	29	DOT	MIDDLE
+ e	30	ARROW	MIDDLE
:	31	COLON	MIDDLE
:=	32	ASSIGNSY	MIDDLE
BEGIN	33	BEGINSY	INITIAL
END	34	ENDSY	TERMINAL
IF	35	IFSY	INITIAL
THEN	36	THENSY	MIDDLE
ELSE	37	ELSESY	TERMINAL
CASE	38	CASESY	INITIAL
OF	39	OFSY	MIDDLE
REPEAT	40	REPEATSY	INITIAL
UNTIL	41	UNTILSY	TERMINAL
WHILE	42	WHILESY	INITIAL
DO	43	DOSY	MIDDLE

00781	T	0375
00782	T	0375
00783	T	0375
00784	T	0375
00785	T	0375
00786	T	0375
00787	T	0375
00788	T	0375
00789	T	0375
00790	T	0375
00791	T	0375
00792	T	0375
00793	T	0375
00794	T	0375
00795	T	0375
00796	T	0375
00797	T	0375
00798	T	0375
00799	T	0375
00800	T	0375
00801	T	0375
00802	T	0375
00803	T	0375
00804	T	0375
00805	T	0375
00806	T	0375
00807	T	0375
00808	T	0375
00809	T	0375
00810	T	0375
00811	T	0375
00812	T	0375
00813	T	0375
00814	T	0375
00815	T	0375
00816	T	0375
00817	T	0375
00818	T	0375
00819	T	0375
00820	T	0375
00821	T	0375
00822	T	0375
00823	T	0375
00824	T	0375
00825	T	0375
00826	T	0375
00827	T	0375
00828	T	0375
00829	T	0375
00830	T	0375
00831	T	0375
00832	T	0375
00833	T	0375
00834	T	0375
00835	T	0375
00836	T	0375
00837	T	0375

```

% FOR 44 FORSY INITIAL
% TO 45 TOSY MIDDLE
% DOWNT0 46 DOWNTOSY MIDDLE
% GOTO 47 GOTOSY INITIAL
% NIL 48 NILSY MIDDLE
% TYPE 49 TYPESY INITIAL
% ARRAY 50 ARRAYSY MIDDLE
% RECORD 51 RECORDSY MIDDLE
% FILE 52 FILESY MIDDLE
% SET 53 SETSY MIDDLE
% CONST 54 CONSTSY INITIAL
% VAR 55 VARSY INITIAL
% LABEL 56 LABELSY INITIAL
% FUNCTION 57 FUNCYSY INITIAL
% PROCEDURE 58 PROCYSY INITIAL
% WITH 59 WITHSY INITIAL
% PROGRAM 60 PROGRAMSY INITIAL
% PACKED 61 PACKEDSY MIDDLE

```

```

% DEFINE BLANK=48#, EQUAL=61#, QUOTES=63#, DOLLAR=42#,%
% LETTER(C)=(17<C AND C<25)OR(33<C AND C<41)OR(50<C AND C<57)#,%
% ALFANUM(C)=(LETTER(C) OR C<9)#;%

```

```

% REAL CURVAL;%
% ALPHA CURNAME1,CURNAME2,C,CX;%
% INTEGER CURLNGTH, LASTCHARPOS;%
% BOOLEAN FINIS;%

```

```

% DEFINE NEXTCHAR=%
% BEGIN COMMENT *** READ NEXT CHARACTER. ***;%
% IF CHARCNT=0 THEN C:=BLANK ELSE%
% BEGIN%
% REPLACE CHARPNT BY CARDPNT;CARDPNT FOR 1;%
% C:=CH[0]; CHARCNT:=CHARCNT-1;%
% END END #;%

```

```

% PROCEDURE INSYMBOL;%
% BEGIN COMMENT *** READS THE NEXT SYMBOL. ***;%
% INTEGER SCALE,EXP;%

```

```

% BOOLEAN NEGEXP;%
% LABEL START,OVERFLOW;%

```

```

% START;%
% IF C=BLANK THEN%
% BEGIN SCAN CARDPNT;CARDPNT FOR CHARCNT;CHARCNT WHILE =" ";%
% IF CHARCNT=0 THEN BEGIN NEWCARD; GO TO START END;%

```

```

NEXTCHAR;%

```

```

00838 T 0375
00839 T 0375
00840 T 0375
00841 T 0375
00842 T 0375
00843 T 0375
00844 T 0375
00845 T 0375
00846 T 0375
00847 T 0375
00848 T 0375
00849 T 0375
00850 T 0375
00851 T 0375
00852 T 0375
00853 T 0375
00854 T 0375
00855 T 0375
00856 T 0375
00857 T 0375
00858 T 0375
00859 T 0375
00860 T 0375
00861 T 0375
00862 T 0375
00863 T 0375
00864 T 0375
00865 T 0375
00866 T 0375
00867 T 0375
00868 T 0375
00869 T 0375
00870 T 0375
00871 T 0375
00872 T 0375
00873 T 0375
00874 T 0375
00875 T 0375
00876 T 0375
00877 T 0375
00878 T 0375
00879 T 0375
START OF SEGMENT ***** 8
00880 T 0000
00881 T 0000
00882 T 0000
00883 T 0000
00884 T 0000
00885 T 0000
00886 T 0003
0000886
0000886
0000886
0000886
0000886
00887 T 0070
0000887

```

END;%	00888 T	0078	00000887
IF LETTER(C) THEN%	00889 T	0078	00000885
BEGIN%	00890 T	0083	
CURLNGTH:=1; CURNAME1:=C; CURNAME2:=0;%	00891 T	0084	
NEXTCHAR;%	00892 T	0086	
			00000892
			00000892
WHILE ALFANUM(C) AND CURLNGTH<6 DO%	00893 T	0094	
BEGIN CURNAME1:=C&CURNAME1[35:29:30];%	00894 T	0102	
CURLNGTH:=CURLNGTH+1; NEXTCHAR;%	00895 T	0104	
			00000895
			00000895
END;%	00896 T	0112	
			00000894
IF CURLNGTH=6 THEN%	00897 T	0113	
BEGIN%	00898 T	0114	
WHILE ALFANUM(C) AND CURLNGTH<12 DO%	00899 T	0114	
BEGIN CURNAME2:=C&CURNAME2[35:29:30];%	00900 T	0123	
CURLNGTH:=CURLNGTH+1; NEXTCHAR;%	00901 T	0125	
			00000901
			00000901
END;%	00902 T	0133	
			00000900
WHILE ALFANUM(C) DO NEXTCHAR;%	00903 T	0134	
			00000903
			00000903
END;%	00904 T	0150	
			00000898
CURNAME1.NAMELENGTH:=CURLNGTH;%	00905 T	0156	
CASE CURLNGTH OF%	00906 T	0152	
BEGIN ;%	00907 T	0152	
CURSY:=IDENTIFIER;%	00908 T	0152	
CURSY:=IF CURNAME1="20000IF" THEN IFSY ELSE%	00909 T	0154	
IF CURNAME1="20000DD" THEN DDSY ELSE%	00910 T	0156	
IF CURNAME1="20000TD" THEN TDSY ELSE%	00911 T	0158	
IF CURNAME1="20000DR" THEN DRSY ELSE%	00912 T	0160	
IF CURNAME1="20000DF" THEN DFSY ELSE%	00913 T	0162	
IF CURNAME1="20000IN" THEN INSY ELSE IDENTIFIER;%	00914 T	0164	
CURSY:=IF CURNAME1="3000END" THEN ENDSY ELSE%	00915 T	0167	
IF CURNAME1="3000FOR" THEN FORSY ELSE%	00916 T	0169	
IF CURNAME1="3000DIV" THEN DIVSY ELSE%	00917 T	0171	
IF CURNAME1="3000MOD" THEN MODSY ELSE%	00918 T	0173	
IF CURNAME1="3000NIL" THEN NILSY ELSE%	00919 T	0175	
IF CURNAME1="3000AND" THEN ANDSY ELSE%	00920 T	0177	
IF CURNAME1="3000NOT" THEN NOTSY ELSE%	00921 T	0179	
IF CURNAME1="3000VAR" THEN VARSY ELSE%	00922 T	0181	
IF CURNAME1="3000SET" THEN SETSY ELSE%	00923 T	0183	
IF CURNAME1="3000LSS" THEN LSSSY ELSE%	00924 T	0185	
IF CURNAME1="3000LEQ" THEN LEQSY ELSE%	00925 T	0187	
IF CURNAME1="3000GEQ" THEN GEQSY ELSE%	00926 T	0189	
IF CURNAME1="3000GTR" THEN GTRSY ELSE%	00927 T	0191	
IF CURNAME1="3000NEQ" THEN NEQSY ELSE%	00928 T	0193	
IF CURNAME1="3000EQL" THEN EQLSY ELSE IDENTIFIER;%	00929 T	0195	
CURSY:=IF CURNAME1="400THEN" THEN THENSY ELSE%	00930 T	0198	
IF CURNAME1="400ELSE" THEN ELSESY ELSE%	00931 T	0200	


```

IF CURNAME1="400WITH" THEN WITHSY ELSE%
IF CURNAME1="400CASE" THEN CASESY ELSE%
IF CURNAME1="400GOTO" THEN GOTOSY ELSE%
IF CURNAME1="400TYPE" THEN TYPESY ELSE%
IF CURNAME1="400FILE" THEN FILES SY ELSE IDENTIFIER;%
CURSY:=IF CURNAME1="50BEGIN" THEN BEGINSY ELSE%
IF CURNAME1="50WHILE" THEN WHILESY ELSE%
IF CURNAME1="50UNTIL" THEN UNTILSY ELSE%
IF CURNAME1="50ARRAY" THEN ARRAYSY ELSE%
IF CURNAME1="50CONST" THEN CONSTSY ELSE%
IF CURNAME1="50LABEL" THEN LABELSY ELSE IDENTIFIER;%
CURSY:=IF CURNAME1="6REPEAT" THEN REPEATSY ELSE%
IF CURNAME1="6DOWNT0" THEN DOWNTOSY ELSE%
IF CURNAME1="6RECORD" THEN RECORDSY ELSE%
IF CURNAME1="6PACKED" THEN PACKEDSY ELSE IDENTIFIER;%
CURSY:=IF CURNAME1="7PROGRA" AND CURNAME2="M" THEN PROGRAMSY%
ELSE IDENTIFIER;%
CURSY:=IF CURNAME1="8FUNCTI" AND CURNAME2="ON" THEN FUNCSY%
ELSE IDENTIFIER;%
CURSY:=IF CURNAME1="9PROCED" AND CURNAME2="UKE" THEN PROCSY%
ELSE IDENTIFIER;%
CURSY:=IDENTIFIER; % 10 CHARACTERS,
CURSY:=IDENTIFIER; % 11 CHARACTERS,
CURSY:=IDENTIFIER; % 12 CHARACTERS.
END OF CASE;%

```

```

00932 T 0202
00933 T 0204
00934 T 0206
00935 T 0208
00936 T 0210
00937 T 0213
00938 T 0215
00939 T 0217
00940 T 0219
00941 T 0221
00942 T 0223
00943 T 0227
00944 T 0229
00945 T 0231
00946 T 0233
00947 T 0236
00948 T 0238
00949 T 0240
00950 T 0243
00951 T 0244
00952 T 0247
00953 T 0249
00954 T 0250
00955 T 0251
00956 T 0252

```

```

00000907
START OF SEGMENT ***** 9
9 IS 13 LONG, NEXT SEG 8

```

```

IF RESWORDOPTION AND CURSY#IDENTIFIER THEN%
BEGIN T1:=CARDLENGTH-CHARCNT-CURLNGTH;%
IF CHARCNT=0 THEN CARDPNT:=CARDPNT+1 ELSE T1:=T1-1;%
REPLACE XLINEPNT+T1 BY CARDPNT-(CURLNGTH+1)%
FOR CURLNGTH;%
END;%

```

```

00957 T 0252
00958 T 0253
00959 T 0255
00960 T 0305
00961 T 0310
00962 T 0313

```

END OF LETTER ELSE%

```

00000958
00963 T 0313

```

```

IF C$9 THEN%
BEGIN%
CURVAL:=C; CURSY:=INTCONST;%
NEXTCHAR;%

```

```

00000890
00964 T 0313
00965 T 0314
00966 T 0314
00967 T 0316

```

```

WHILE C$9 DO BEGIN CURVAL:=10×CURVAL+C; NEXTCHAR END;%

```

```

00000967
00000967
00968 T 0323

```

```

IF C="." THEN%
BEGIN%
NEXTCHAR;%

```

```

00000968
00000968
00000968
00969 T 0335
00970 T 0335
00971 T 0336

```

```

IF C$9 THEN%
BEGIN CURSY:=REALCONST;%
DO BEGIN CURVAL:=10×CURVAL+C;%
SCALE:=SCALE-1; NEXTCHAR;%

```

```

00000971
00000971
00972 T 0343
00973 T 0344
00974 T 0345
00975 T 0347

```

00000975


```

IF NOT FINIS THEN%
BEGIN%
  REPLACE STRINGPNT+CURLNGTH BY CHARPNT FOR 1;%
  CURLNGTH:=CURLNGTH+1;%
  NEXTCHAR;%

END END UNTIL FINIS;%

IF CURLNGTH=0 THEN ERROR(4) ELSE%
IF CURLNGTH=1 THEN%
BEGIN CURSY:=CHARCONST;%
  REPLACE CHARPNT BY STRINGPNT FOR 1; CURVAL:=CH[0];%
END ELSE%

IF CURLNGTH<7 THEN%
BEGIN TEXT[0]:="      "%;
  REPLACE TEXTPNT BY STRINGPNT FOR CURLNGTH;%
  CURVAL:=TEXT[0];%
END;%

END OF STRINGS ELSE%

BEGIN%
  CURSY:=SYMBOL[C]; NEXTCHAR;%

IF CURSY=COLON AND C=EQUAL THEN%
  BEGIN CURSY:=ASSIGNSY; NEXTCHAR END ELSE%

IF CURSY=DOT AND C="," THEN%
  BEGIN CURSY:=DOUBLEDOT; NEXTCHAR END ELSE%

IF CURSY=LSSSY AND C=EQUAL THEN%
  BEGIN CURSY:=LEQSY; NEXTCHAR END ELSE%

IF CURSY=LSSSY AND C=">" THEN%
  BEGIN CURSY:=NEQSY; NEXTCHAR END ELSE%

IF CURSY=GTRSY AND C=EQUAL THEN%
  BEGIN CURSY:=GEQSY; NEXTCHAR END ELSE%

IF CURSY=LPAR AND C="*" THEN%
BEGIN
  NEXTCHAR;%

```

```

00001008
01009 T 0444
01010 T 0444
01011 T 0445
01012 T 0450
01013 T 0451
00001013
00001013
01014 T 0458
00001010
00001006
01015 T 0459
01016 T 0461
01017 T 0462
01018 T 0464
01019 T 0467
00001017
01020 T 0467
01021 T 0469
01022 T 0470
01023 T 0473
01024 T 0474
00001021
01025 T 0474
00001003
01026 T 0474
01027 T 0476
00001027
00001027
01028 T 0484
01029 T 0486
00001029
00001029
00001029
01030 T 0495
01031 T 0497
00001031
00001031
00001031
01032 T 0506
01033 T 0508
00001033
00001033
00001033
01034 T 0517
01035 T 0519
00001035
00001035
00001035
01036 T 0528
01037 T 0530
00001037
00001037
00001037
01038 T 0539
01039 T 0541
01040 T 0541

```

```

% *** COMMENT ***

```

			00001040
			00001040
IF C=DOLLAR THEN	% DOLLAR INDICATES COMPILER OPTIONS,	01041 T	0549
DO BEGIN%		01042 T	0550
NEXTCHAR; CX:=C; NEXTCHAR;%		01043 T	0551
			00001043
			00001043
			00001043
			00001043
IF CX="L" THEN IF C=1 THEN HEADING%		01044 T	0566
			00001044
	ELSE LISTOPTION:=C="+"	01045 T	0590
IF CX="R" THEN RESWORDOPTION:=C="+"	ELSE%	01046 T	0592
IF CX="C" THEN CHECKOPTION:=C="+"	ELSE%	01047 T	0595
IF CX="D" THEN DUMPOPTION:=C="+"	ELSE%	01048 T	0598
IF CX="X" THEN XREFOPTION:=C="+"	ELSE%	01049 T	0601
IF CX="A" THEN%		01050 T	0604
IF C="+"	THEN WRITE(PASCALGOL,ALIST)%	01051 T	0605
	ELSE WRITE(PASCALGOL,NOALIST) ELSE%	01052 T	0609
IF CX="T" THEN%		01053 T	0613
BEGIN LASTCHARPOS := CHARCNT - CARDLLENGTH;%		01054 T	0614
CARDLENGTH:=10xC;%		01055 T	0616
NEXTCHAR; CARDLLENGTH:=CARDLLENGTH+C;%		01056 T	0617
			00001056
			00001056
IF CARDLLENGTH<9 OR CARDLLENGTH>80 THEN%		01057 T	0626
BEGIN ERROR(14); CARDLLENGTH:=72 END;%		01058 T	0628
			00001058
CHARCNT:=MAX(0,LASTCHARPOS+CARDLLENGTH-1);%		01059 T	0630
END;%		01060 T	0634
			00001054
NEXTCHAR;%		01061 T	0634
			00001061
			00001061
END UNTIL C=",";%		01062 T	0641
			00001042
FINIS:=FALSE;%		01063 T	0643
DO BEGIN%		01064 T	0643
IF C="+" THEN%		01065 T	0644
SCAN CARDPNT;CARDPNT FOR CHARCNT;CHARCNT UNTIL "+";%		01066 T	0644
IF CHARCNT=0 THEN NEWCARD ELSE%		01067 T	0647
			00001067
			00001067
			00001067
			00001067
BEGIN NEXTCHAR;%		01068 T	0714
			00001068
			00001068
WHILE C="+" DO NEXTCHAR;%		01069 T	0722
			00001069
			00001069
FINIS:=C=")";%		01070 T	0731
END END UNTIL FINIS;%		01071 T	0732
			00001068
			00001064
NEXTCHAR;%		01072 T	0733
			00001072

GO TO START;%
END OF COMMENT;%
END;%
END OF INSYMBOL;%

01073 T 0740 00001072
01074 T 0741 00001039
01075 T 0741 00001026
01076 T 0741 00000878
8 IS 753 LONG, NEXT SEG 2

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
%
%
%
%
%
%
%
%
%
%
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
%
%
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 WRITEXPR=%
BEGIN%
  FOR T1:=1 STEP 1 UNTIL NUMSYMS DO%
    BEGIN%
      SX:=SYMTAB[T1]; TX:=SX.[41:6];%

```

```

01078 T 0375
01079 T 0375
01080 T 0375
01081 T 0375
01082 T 0375
01083 T 0375
01084 T 0375
01085 T 0375
01086 T 0375
01087 T 0375
01088 T 0375
01089 T 0375
01090 T 0375
01091 T 0375
01092 T 0375
01093 T 0375
01094 T 0375
01095 T 0375
01096 T 0375
01097 T 0375
01098 T 0375
01099 T 0375
01100 T 0375
01101 T 0375
01102 T 0375
01103 T 0375
01104 T 0375
01105 T 0375
01106 T 0375
01107 T 0375
01108 T 0375
01109 T 0375
01110 T 0375
01111 T 0375
01112 T 0375
01113 T 0375
01114 T 0375
01115 T 0375
01116 T 0375
01117 T 0375
01118 T 0375
01119 T 0375
01120 T 0375
01121 T 0375
01122 T 0375
01123 T 0375
01124 T 0375
01125 T 0375
01126 T 0375
01127 T 0375
01128 T 0375
01129 T 0375
01130 T 0375
01131 T 0375
01132 T 0375
01133 T 0375
01134 T 0375

```

```

IF TX=0 THEN GEN(SX,7,2) ELSE%
IF TX=3 THEN ELSE%
IF TX=1 THEN GEN(SX,1,7) ELSE%
BEGIN%
  T1:=T1+1; SX:=SYMTAB[T1];%
  IF SX.[4416]=0 THEN GENINT(SX) ELSE GENREAL(SX);%
END END;%
NUMSYMS:=0;%
END OF WRITEEXPR#;%
%
%
DEFINE CHECKEXPR(LLIM,ULIM)=%
BEGIN%
  PUTTEXT("CHECK(");%
  EXPRESSION;%
  PUTSYM(","); PUTCONST(LLIM);%
  PUTSYM(","); PUTCONST(ULIM);%
  PUTSYM(","); PUTCONST(CARDCNT);%
  PUTSYM(")");%
END OF CHECKEXPR#;%
%
%
BOOLEAN SIMPLEVARIABLE,INSIDEBRACKETS;%
INTEGER NUMPOINTERS;%
%
PROCEDURE VARIABLE;%
BEGIN%
  INTEGER STARTSYM,LLIM,ULIM;%

  REAL T;%
  BOOLEAN INBRACKET,INRECORD;%
  LABEL ADDADDR;%
%
  STARTSYM:=NUMSYMS+1;%
  IF THISLEVEL>CURLEVEL THEN % VARIABLE IN FIELD LIST OF
  BEGIN % RECORD USED IN WITH=STATEMENT.
    T:=DISPLAY[THISLEVEL];%
    T4:=T,FIRSTWITHSYM; T5:=T,LASTWITHSYM;%
    FOR T3:=T4 STEP 1 UNTIL T5 DO PUTTEXT(WITHTAB[T3]);%

    INRECORD:=TRUE;%
    INBRACKET:=BOOLEAN(T,BRACKETSINWITH);%
    NUMPOINTERS:=NUMPOINTERS+T,NUMPNTRSINWITH;%
    SIMPLEVARIABLE:=FALSE;%
    CURTYPE:=T,RECTYPE; T:=TYPETAB1[CURTYPE];%
    GO TO ADDADDR;%
  END;%

  IF THISLEVEL>1 AND THISLEVEL<CURLEVEL THEN ERROR(5);%
  CURTYPE:=THISID.TYPE; SIMPLEVARIABLE:=TRUE;%
  PUTID("V",1000*THISLEVEL+THISINDEX,5);%

  INSYMBOL;%
  IF CURSY=LBACKET OR CURSY=DOT OR CURSY=ARROW THEN%

```

```

01135 T 0375
01136 T 0375
01137 T 0375
01138 T 0375
01139 T 0375
01140 T 0375
01141 T 0375
01142 T 0375
01143 T 0375
01144 T 0375
01145 T 0375
01146 T 0375
01147 T 0375
01148 T 0375
01149 T 0375
01150 T 0375
01151 T 0375
01152 T 0375
01153 T 0375
01154 T 0375
01155 T 0375
01156 T 0375
01157 T 0375
01158 T 0375
01159 T 0375
01160 T 0375
01161 T 0375
01162 T 0375
START OF SEGMENT ***** 10
01163 T 0000
01164 T 0000
01165 T 0000
01166 T 0000
01167 T 0000
01168 T 0001
01169 T 0002
01170 T 0002
01171 T 0003
01172 T 0006
00001172
00001172
01173 T 0015
01174 T 0016
01175 T 0017
01176 T 0019
01177 T 0019
01178 T 0022
01179 T 0022
00001169
01180 T 0022
01181 T 0025
01182 T 0027
00001182
00001182
00001182
01183 T 0043
01184 T 0044

```



```

CURTYPE:=T,ARRTYPE)%
IF INRECORD THEN%
BEGIN%
  IF LLIM<0 THEN BEGIN PUTSYM("+"); PUTCONST(-LLIM) END ELSE

  IF LLIM>0 THEN BEGIN PUTSYM("-"); PUTCONST( LLIM) END;%

  PUTSYM(")");%

  IF TYPETAB1[CURTYPE].SIZE>1 THEN%
  BEGIN PUTSYM("x"); PUTCONST(TYPETAB1[CURTYPE].SIZE) END;%

  END ELSE IF TYPETAB1[CURTYPE].STRUCT>0 THEN PUTSYM(",");%

  END UNTIL CURSY#COMMA;%

  IF CURSY#RBRACKET THEN%
  BEGIN ERROR(59); SKIP(RBRACKET);%
    IF CURSY=RBRACKET THEN INSYMBOL;%
  END ELSE INSYMBOL;%

  END OF BRACKETS ELSE%

  IF CURSY=DOT THEN%
  BEGIN%
    IF NOT(INBRACKET OR INRECORD) THEN%
    BEGIN PUTSYM("["); INBRACKET:=TRUE END;%

```

```

00001199
01200 T 0212
01201 T 0213
01202 T 0213
01203 T 0214
00001203
00001203
00001203
00001203
00001203
00001203
00001203
00001203
00001203
01204 T 0235
00001204
00001204
00001204
00001204
00001204
00001204
00001204
00001204
00001204
01205 T 0257
00001205
00001205
00001205
01206 T 0265
01207 T 0266
00001207
00001207
00001207
00001207
00001207
00001207
00001207
00001207
00001207
01208 T 0288
00001202
00001208
00001208
00001208
01209 T 0298
00001192
01210 T 0299
01211 T 0300
01212 T 0302
01213 T 0304
00001211
01214 T 0305
00001189
01215 T 0305
01216 T 0306
01217 T 0306
01218 T 0307

```

```

T:=TYPETAB1[CURTYPE];%
IF T.FORM#RECORD THEN ERROR(12);%
INSYMBOL;%
IF CURSY#IDENTIFIER THEN%
BEGIN%
  SEARCHTAB(T,RECTAB);%

  IF FOUND THEN%
  BEGIN%
    THISID:=NAMETAB3(T,RECTAB,THISINDEX);%
    PUTSYM("+");%
  END%

  PUTCONST(THISID,INFO); CURTYPE:=THISID,TYPE;%

  END ELSE BEGIN ERROR(1); CURTYPE:=0 END;%

  END ELSE BEGIN ERROR(9); CURTYPE:=0 END;%

  INRECORD:=TRUE;%
  INSYMBOL;%
  END OF DOT ELSE%

  BEGIN % CURSY=ARROW
  T:=TYPETAB1[CURTYPE];%
  IF T.FORM#FILES THEN%
  BEGIN%
    CURTYPE:=T,FILETYPE;%
    IF TYPETAB1[CURTYPE].STRUCT#0 THEN PUTTEXT(" [0]");%

  END ELSE%

  IF T.FORM#TEXTFILE THEN%
  BEGIN%
    SYMTAB[NUMSYMS]:=SYMTAB[NUMSYMS] & "I" [35:5:6];%
    PUTSYM(","); PUTTEXT("LASTCH");%

  END ELSE%

  CURTYPE:=CHARTYPE;%
  END ELSE%

```

```

00001218
00001218
00001218
00001218
01219 T 0316
01220 T 0317
01221 T 0320
01222 T 0320
01223 T 0321
01224 T 0321
00001224
00001224
01225 T 0342
01226 T 0342
01227 T 0342
01228 T 0345
00001228
00001228
00001228
01229 T 0352
00001229
00001229
00001229
00001229
00001229
01230 T 0367
00001226
00001230
01231 T 0369
00001223
00001231
01232 T 0371
01233 T 0372
01234 T 0372
00001216
01235 T 0372
01236 T 0373
01237 T 0374
01238 T 0375
01239 T 0375
01240 T 0377
00001240
00001240
01241 T 0384
00001238
01242 T 0384
01243 T 0388
01244 T 0388
01245 T 0391
00001245
00001245
00001245
00001245
00001245
01246 T 0404
01247 T 0405
00001243

```

```

IF T.FORM=POINTERS THEN%
BEGIN%
  IF INBRACKET THEN PUTSYM("]");%

  INBRACKET:=FALSE;%
  IF NUMSYMS+2<MAXSYMS THEN%
  BEGIN%
    FOR T1:=NUMSYMS STEP -1 UNTIL STARTSYM DO%
      SYMTAB[T1+2]:=SYMTAB[T1];%
      SYMTAB[STARTSYM]:=MEM["];%
      SYMTAB[STARTSYM+1]:= (T:=");%
      NUMSYMS:=NUMSYMS+2; NUMPOINTERS:=NUMPOINTERS+1;%
      INRECORD:=TRUE;%
    END ELSE ERROR(63);%

    CURTYPE:=T.POINTTYPE;%
    END ELSE BEGIN ERROR(12); CURTYPE:=0 END;%

    INSYMBOL;%
    END OF ARROW;%

  END UNTIL CURSY<LBRACKET AND CURSY<DOT AND CURSY<ARROW;%

  IF TYPETAB1[CURTYPE].STRUCT=0 THEN%
  BEGIN%
    IF INBRACKET THEN PUTSYM("]");%

    WHILE NUMPOINTERS>0 DO%
      BEGIN PUTTEXT("=1)DIV"); PUTTEXT(" 1022,");%

      PUTTEXT(" T MOD"); PUTTEXT(" 1022]");%

      NUMPOINTERS:=NUMPOINTERS-1;%
    END;%
  END;%
END;%

INSIDEBRACKETS:=INBRACKET;%
CURMODE:=NUMBER;%
END OF VARIABLE;%

```

```

01248 T 0405
01249 T 0408
01250 T 0408
00001250
00001250
00001250
01251 T 0417
01252 T 0417
01253 T 0419
01254 T 0419
01255 T 0421
01256 T 0425
01257 T 0426
01258 T 0428
01259 T 0430
01260 T 0431
00001253
01261 T 0434
01262 T 0438
00001249
00001262
01263 T 0438
01264 T 0438
00001235
01265 T 0438
00001187
01266 T 0441
01267 T 0443
01268 T 0443
00001268
00001268
00001268
01269 T 0452
01270 T 0453
00001270
00001270
00001270
00001270
01271 T 0466
00001271
00001271
00001271
00001271
01272 T 0480
01273 T 0481
00001270
01274 T 0484
00001267
01275 T 0484
00001185
01276 T 0484
01277 T 0484
01278 T 0485
00001161

```

```

%
%
PROCEDURE PASSPARAMS;%
BEGIN%
  INTEGER NPARS,PARAM,PTYPE,P,FIRSTSYM;%

  BOOLEAN FORMALPROC,CHECK;%
  LABEL EXIT;%

%
  PUTID("V",1000*THISLEVEL+THISINDEX,5);%

  P:=THISID.INFO;%
  FORMALPROC:=BOOLEAN(THISID.FORMAL);%
  NPARS:=PARAMTAB[P]; P:=P+1;%
  IF FORMALPROC THEN NPARS:=9999;%
  INSYMBOL;%
  IF CURSY=LPAR THEN%
  BEGIN%
    PUTSYM("(");%

    DO BEGIN%
      INSYMBOL;%
      IF NPARS=0 THEN BEGIN ERROR(3); SKIP(RPAR); GO TO EXIT END;%

      PARAM:=PARAMTAB[P]; P:=P+1;%
      PTYPE:=PARAM.PARAMTYPE;%
      IF PARAM.PARAMKIND=CONST THEN%
      BEGIN%
        CHECK:=CHECKOPTION AND TYPETAB1[PTYPE],FORM LEQ CHAR;%
        IF CHECK THEN PUTTEXT("CHECK(");%

        PUTDUMMY; FIRSTSYM:=NUMSYMS;%

        EXPRLEVEL:=EXPRLEVEL+1;%
        EXPRESSION; EXPRLEVEL:=EXPRLEVEL-1;%
        IF CURMODE=BITPATTERN THEN%
        BEGIN SYMTAB[FIRSTSYM]:=" REAL("; PUTSYM(")"; END;%

        IF CHECK THEN%
        BEGIN%
          PUTSYM(","); PUTCONST(TYPETAB2[PTYPE]);%

```

```

01279 T 0375
01280 T 0375
01281 T 0375
01282 T 0375
01283 T 0375
START OF SEGMENT ***** 11
01284 T 0000
01285 T 0000
01286 T 0000
01287 T 0000
00001287
00001287
00001287
01288 T 0016
01289 T 0018
01290 T 0019
01291 T 0021
01292 T 0023
01293 T 0023
01294 T 0024
01295 T 0024
00001295
00001295
00001295
01296 T 0033
01297 T 0034
01298 T 0034
00001298
01299 T 0037
01300 T 0040
01301 T 0041
01302 T 0042
01303 T 0043
01304 T 0045
00001304
00001304
01305 T 0052
00001305
00001305
00001305
01306 T 0060
01307 T 0061
01308 T 0063
01309 T 0064
00001309
00001309
00001309
00001309
01310 T 0075
01311 T 0075
01312 T 0076
00001312
00001312
00001312
00001312
00001312
00001312

```

PUTSYM(","); PUTCONST(TYPETAB3[PTYPE]);%

00001312
00001312
01313 T 0096
00001318
00001319
00001319
00001313
00001312
00001313
00001313
00001313
00001313

PUTSYM(","); PUTCONST(CARDCNT); PUTSYM(",");%

01314 T 0117
00001314
00001314
00001314
00001314
00001314
00001314
00001314
00001314
00001314
00001314

END;%

01315 T 0146
00001311

END ELSE%

01316 T 0146
00001302

IF PARAM.PARAMKIND=VAR THEN%

BEGIN%

IF CURSY=IDENTIFIER THEN%

BEGIN%

SEARCH;%

01317 T 0146
01318 T 0147
01319 T 0148
01320 T 0149
01321 T 0149

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(",");%

00001321
00001321
00001321
00001321

01322 T 0178
01323 T 0178
01324 T 0178
01325 T 0180
01326 T 0182
01327 T 0183
01328 T 0184
01329 T 0184
01330 T 0186

PUTID("F",1000*THISLEVEL+THISINDEX,5); PUTSYM(",");%

00001330
00001330
00001330
00001330
00001330
00001330
01331 T 0210
00001331
00001331
00001331
00001331
00001331

```

        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);%

```

```

01332 T 0234 00001331
00001332
00001332
00001332
01333 T 0250
01334 T 0251 00001328
01335 T 0251
01336 T 0251
01337 T 0252
01338 T 0253
01339 T 0256
01340 T 0256 00001335
00001326
00001340
01341 T 0258 00001323
00001341
01342 T 0260 00001320
00001342
01343 T 0262 00001318
01344 T 0262
01345 T 0262
01346 T 0263
01347 T 0263 00001347
00001347
00001347
00001347
01348 T 0292
01349 T 0292
01350 T 0292
01351 T 0295 00001351
00001351
00001351
01352 T 0312
01353 T 0316
01354 T 0317 00001349
00001354
01355 T 0319 00001348
00001355
01356 T 0321 00001344
01357 T 0321 00001357
00001357
00001357
00001357
00001357
00001357
00001357

```

```

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;

```

```

00001357
01358 T 0380
01359 T 0381
00001359
00001359
00001359
01360 T 0390
00001298
01361 T 0391
00001361
01362 T 0394
00001362
00001362
00001362
01363 T 0401
01364 T 0403
00001294
01365 T 0403
01366 T 0406
01367 T 0407
00001282
11 IS 414 LONG, NEXT SEG 2

```

```

%
%
PROCEDURE FACTOR;
BEGIN
INTEGER STARTSYM,STYPE,T;

BOOLEAN FIRST;
REAL VAL;

%
DEFINE PARAMETER= %*** CHECK THAT THE FUNCTION HAS 1 PARAM.
BEGIN%
INSYMBOL;
IF CURSY=LPAR THEN%
BEGIN%
PUTSYM("("); INSYMBOL; EXPRESSION;
IF TYPETAB1[CURTYPE].FORM=NUMERIC THEN CURTYPE:=INTTYPE;
IF CURSY#RPAR THEN BEGIN ERROR(3); SKIP(RPAR) END;
PUTSYM(")"); IF CURSY=RPAR THEN INSYMBOL;
END ELSE ERROR(3);
END OF PARAMETER#;

%
CURMODE:=NUMBER;
IF CURSY=IDENTIFIER THEN%
BEGIN%
SEARCH;

IF FOUND THEN%
BEGIN%

```

```

%*** FACTOR ***
%*****

```

```

01368 T 0375
01369 T 0375
01370 T 0375
01371 T 0375
01372 T 0375
START OF SEGMENT ***** 12
01373 T 0000
01374 T 0000
01375 T 0000
01376 T 0000
01377 T 0000
01378 T 0000
01379 T 0000
01380 T 0000
01381 T 0000
01382 T 0000
01383 T 0000
01384 T 0000
01385 T 0000
01386 T 0000
01387 T 0000
01388 T 0000
01389 T 0000
01390 T 0001
01391 T 0002
00001391
00001391
00001391
00001391
01392 T 0031
01393 T 0031

```



```

IF TYPETAB1[CURTYPE].FORM#NUMERIC THEN ERROR(67))%
IF CURSY#RPAR THEN BEGIN ERROR(46)); SKIP(RPAR) END;%

      IF CURSY=RPAR THEN INSYMBOL;%
END ELSE ERROR(58))%

CURTYPE:=CHARTYPE;%
END ELSE%

IF CURNAME1="3000EOF" OR                                     % "EOF"/"EOLN"
CURNAME1="400EOLN" THEN%
BEGIN%
FIRST:=CURNAME1="3000EOF";%
FILEPARAM(INPUTFILE);%

      PUTID("I",FILENAME,5));%

      PUTTEXT(IF FIRST THEN " ,EOF" ELSE " ,EOLN"));%

IF LPARFOUND THEN%
BEGIN%
      IF CURSY#RPAR THEN BEGIN ERROR(46)); SKIP(RPAR) END;%

```

```

00001418
00001418
00001418
00001418
00001418
00001418
00001418
00001418
00001418
00001418
00001418
00001418
00001418
00001418
00001418
00001418
00001418
00001418
00001418
00001418
01419 T 0120
01420 T 0122
00001420
01421 T 0125
01422 T 0127
00001418
01423 T 0128
01424 T 0129
00001415
01425 T 0129
01426 T 0130
01427 T 0131
01428 T 0132
01429 T 0133
00001429
00001429
00001429
00001429
00001429
00001429
00001429
00001429
00001429
00001429
01430 T 0186
00001430
00001430
00001430
01431 T 0201
00001431
00001431
01432 T 0209
01433 T 0209
01434 T 0209
00001434

```

```

        IF CURSY=RPAR THEN INSYMBOL;%
        END;%

        CURTYPE:=BOOLTYPE;%
        END ELSE%

        IF CURNAME1="3000DD" THEN                % "DD"
        BEGIN%
        PUTTEXT("  ODD"); PARAMETER;%

        IF CURTYPE≠INTTYPE THEN ERROR(67);%
        CURTYPE:=BOOLTYPE; CURMODE:=BITPATTERN;%
        END ELSE%

        IF CURNAME1="3000DD" THEN                % "DD"
        BEGIN%
        PUTSYM("("); INSYMBOL;%

        IF CURSY=LPAR THEN%
        BEGIN%
        INSYMBOL; EXPRESSION;%
        IF TYPETAB1[CURTYPE].FORM>CHAR THEN ERROR(67);%
        IF CURSY=RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;%

        INSYMBOL;%
        END ELSE ERROR(58);%

        CURTYPE:=INTTYPE; PUTSYM(")");%

        END ELSE%

        IF CURNAME1="400PRED" OR                  % "PRED"/"SUCC"
        CURNAME1="400SUCC" THEN%
        BEGIN%
        FIRST:=CURNAME1="400PRED";%
        PUTTEXT("CHECK("); INSYMBOL;%

        IF CURSY=LPAR THEN%
        BEGIN%
        INSYMBOL; EXPRESSION;%
        PUTSYM(IF FIRST THEN "-" ELSE "+"); PUTSYM("1");%

```

```

01435 T 0212
01436 T 0214
00001439
01437 T 0214
01438 T 0215
00001427
01439 T 0215
01440 T 0218
01441 T 0219
00001441
00001441
00001441
00001441
00001441
00001441
00001441
00001441
00001441
00001441
00001441
00001441
01442 T 0254
01443 T 0256
01444 T 0258
00001440
01445 T 0258
01446 T 0259
01447 T 0259
00001447
00001447
00001447
01448 T 0269
01449 T 0269
01450 T 0270
01451 T 0271
01452 T 0274
00001452
01453 T 0276
01454 T 0277
00001449
01455 T 0278
00001455
00001455
00001455
01456 T 0286
00001446
01457 T 0286
01458 T 0288
01459 T 0289
01460 T 0289
01461 T 0290
00001461
00001461
01462 T 0299
01463 T 0299
01464 T 0300
01465 T 0301
00001465

```


			00001492
			00001492
			00001492
			00001492
			00001492
			00001492
NUMSYMS:=NUMSYMS-1; PUTSYM(",");%	01493 T	0573	
			00001493
			00001493
			00001493
PUTCONST(CARDCNT); PUTSYM(")");%	01494 T	0582	
			00001494
			00001494
			00001494
			00001494
			00001494
			00001494
			00001494
			00001494
IF CURTYPE≠REALTYPE THEN ERROR(67);%	01495 T	0603	
CURTYPE:=INTTYPE;%	01496 T	0605	
END ELSE%	01497 T	0605	
			00001491
IF CURNAME1="6CONCAT" THEN % "CONCAT"	01498 T	0605	
CONCAT ELSE%	01499 T	0607	
IF CURNAME1="400TIME" THEN % "TIME"	01500 T	0608	
BEGIN%	01501 T	0610	
PUTTEXT("(TIME("); PUTTEXT("1)/60)");%	01502 T	0611	
			00001502
			00001502
			00001502
			00001502
CURTYPE:=REALTYPE; INSYMBOL;%	01503 T	0625	
END ELSE%	01504 T	0626	
			00001501
IF CURNAME1="400DATE" THEN % "DATE"	01505 T	0626	
BEGIN%	01506 T	0629	
PUTTEXT("CURDAT");%	01507 T	0630	
			00001507
			00001507
CURTYPE:=ALFATYPE; INSYMBOL;%	01508 T	0637	
END ELSE%	01509 T	0638	
			00001506
IF CURNAME1="7ELAPSE" AND CURNAME2="D" THEN % "ELAPSED"	01510 T	0638	
BEGIN%	01511 T	0642	
PUTTEXT("(TIME("); PUTTEXT("2)/60)");%	01512 T	0643	
			00001512
			00001512
			00001512
			00001512
CURTYPE:=REALTYPE; INSYMBOL;%	01513 T	0657	
END ELSE%	01514 T	0658	
			00001511
IF CURNAME1="6IOTIME" THEN % "IOTIME"	01515 T	0658	
BEGIN%	01516 T	0661	
PUTTEXT("(TIME("); PUTTEXT("3)/60)");%	01517 T	0662	
			00001517

```

CURTYPE:=REALTYPE; INSYMBOL;%
END ELSE%

IF CURNAME1="7WEEKDA" AND CURNAME2="Y" THEN % "WEEKDAY"
BEGIN%
PUTTEXT("WEEKDA");%

CURTYPE:=ALFATYPE; INSYMBOL;%
END ELSE IF CURNAME1="400USER" THEN % "USER"

BEGIN%
PUTTEXT(" TIME"); PUTTEXT(" (=1)");%

CURTYPE:=ALFATYPE; INSYMBOL;%
END ELSE % "SIN", "COS" ETC,

BEGIN%
PUTTEXT(IF CURNAME1="3000SIN" THEN " SIN" ELSE%
IF CURNAME1="3000COS" THEN " COS" ELSE%
IF CURNAME1="6ARCTAN" THEN "ARCTAN" ELSE%
IF CURNAME1="400SQRT" THEN " SQRT" ELSE%
IF CURNAME1="3000EXP" THEN " EXP" ELSE%
" LN");%

PARAMETER;%

IF CURTYPE#REALTYPE AND CURTYPE#INTTYPE THEN ERROR(67);%
CURTYPE:=REALTYPE;%
END;%

END OF INTRINSIC FUNCTIONS ELSE%

BEGIN%
T:=THISID,TYPE;%
PASSPARAMS;%
CURTYPE:=T;%
END;%

END OF FUNCTIONS ELSE%

```

```

00001517
00001517
00001517
01518 T 0676
01519 T 0677
00001516
01520 T 0677
01521 T 0681
01522 T 0682
00001522
00001522
01523 T 0689
01524 T 0690
00001521
01525 T 0693
01526 T 0694
00001526
00001526
00001526
00001526
01527 T 0708
01528 T 0709
00001525
01529 T 0709
01530 T 0712
01531 T 0712
01532 T 0712
01533 T 0712
01534 T 0712
01535 T 0712
00001539
00001535
01536 T 0727
00001536
00001536
00001536
00001536
00001536
00001536
00001536
00001536
00001536
00001536
00001536
01537 T 0765
01538 T 0768
01539 T 0769
00001529
01540 T 0769
00001407
13 IS 784 LONG, NEXT SEG 12
01541 T 0076
01542 T 0076
01543 T 0077
01544 T 0078
01545 T 0079
00001541
01546 T 0079
00001405

```

```

IF THISID.IDCLASS=PROC THEN%
BEGIN%
  ERROR(68); PASSPARAMS;%
  CURTYPE:=0;%
  END ELSE BEGIN ERROR(69); CURTYPE:=0; INSYMBOL END;%

END ELSE BEGIN ERROR(1); CURTYPE:=0; INSYMBOL END;%

END OF IDENTIFIER ELSE%

IF CURSY≠CHARCONST THEN%
BEGIN%
  CONSTANT(VAL,CURTYPE); PUTCONST(VAL);%

END ELSE%

IF CURSY=NOTSY THEN%
BEGIN%
  PUTTEXT(" NOT "); PUTDUMMY; STARTSYM:=NUMSYMS;%

INSYMBOL; FACTOR;%
IF CURTYPE>0 THEN%
IF CURTYPE≠BOOLTYPE THEN BEGIN ERROR(17); CURTYPE:=0 END;%

IF CURMODE=NUMBER THEN%
BEGIN SYMTAB[STARTSYM]:=" B("; PUTSYM(")");%

```

```

01547 T 0079
01548 T 0080
01549 T 0081
01550 T 0082
01551 T 0083
    00001548
    00001551
01552 T 0085
    00001393
    00001552
01553 T 0088
    00001390
01554 T 0088
01555 T 0089
01556 T 0090
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
    00001556
01557 T 0190
    00001555
01558 T 0190
01559 T 0191
01560 T 0192
    00001560
    00001560
    00001560
    00001560
    00001560
01561 T 0206
01562 T 0207
01563 T 0208
    00001563
01564 T 0211
01565 T 0212
    00001565
    00001565

```

```

        CURMODE:=BITPATTERN;%
    END;%
END ELSE%
IF CURSY=NILSY THEN%
BEGIN%
    PUTCONST(0); CURTYPE:=NILTYPE;%

```

```

        INSYMBOL;%
    END ELSE%
IF CURSY=LPAR THEN%
BEGIN%
    PUTSYM("(");%

```

```

        INSYMBOL; EXPRESSION;%
    IF CURSY≠RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;%
    PUTSYM(")");%

```

```

        INSYMBOL;%
    END ELSE%

```

```

IF CURSY=LBRACKET THEN                                %*** SET CONSTANT ***
BEGIN%
    INSYMBOL;%
    IF CURSY=RBRACKET THEN%
    BEGIN%
        PUTCONST(0); CURTYPE:=EMPTYSET; CURMODE:=NUMBER;%
    
```

```

        INSYMBOL;%
    END ELSE%
BEGIN%
    FIRST:=TRUE;%
    DO BEGIN%
        IF FIRST THEN FIRST:=FALSE ELSE INSYMBOL;%
        PUTTEXT(" BIT("); STARTSYM:=NUMSYMS;%
    
```

```

        EXPRESSION;%
    IF STYPE=0 THEN%
    BEGIN STYPE:=CURTYPE;%

```

```

01566 T 0223 00001565
01567 T 0224
01568 T 0224 00001565
01569 T 0224 00001559
01570 T 0225
01571 T 0226
00001571
00001571
00001571
00001571
00001571
01572 T 0239
01573 T 0239
00001570
01574 T 0239
01575 T 0241
01576 T 0241
00001576
00001576
00001576
01577 T 0249
01578 T 0250
00001578
01579 T 0252
00001579
00001579
00001579
01580 T 0260
01581 T 0260
00001575
01582 T 0260
01583 T 0262
01584 T 0262
01585 T 0263
01586 T 0263
01587 T 0264
00001587
00001587
00001587
00001587
00001587
01588 T 0279
01589 T 0279
00001586
01590 T 0279
01591 T 0280
01592 T 0280
01593 T 0281
01594 T 0283
00001594
00001594
01595 T 0290
01596 T 0290
01597 T 0291

```



```

        IF TYPETAB1[CURTYPE].FORM>CHAR THEN ERROR(72);%
    END ELSE CHECKTYPES(STYPE,CURTYPE);%

```

```

    IF CURSY=DOUBLEDOT THEN%
    BEGIN%
        PUTSYM(","); SYMTAB[STARTSYM]:= " BITS(";%

```

```

        INSYMBOL; EXPRESSION;%
        IF STYPE=0 THEN%
        BEGIN STYPE:=CURTYPE;%
            IF TYPETAB1[CURTYPE].FORM>CHAR THEN ERROR(72);%
        END ELSE CHECKTYPES(STYPE,CURTYPE);%

```

```

    END;%
    PUTSYM(","); PUTCONST(CARDCNT); PUTSYM(")");%

```

```

        IF CURSY=COMMA THEN PUTTEXT(" OR");%

```

```

    END UNTIL CURSY≠COMMA;%

```

```

    IF CURSY≠RBRACKET THEN%
    BEGIN ERROR(59); SKIP(RBRACKET);%
        IF CURSY=RBRACKET THEN INSYMBOL;%
    END ELSE INSYMBOL;%

```

```

    NEWTYPE; T1:=SET; T1.SIZE:=1; T1.STRUCT:=0;%

```

```

    T1.SETTYPE:=STYPE; TYPETAB1[TYPEINDEX]:=T1;%

```

```

01598 T 0292
01599 T 0295

```

```

00001597
00001599
00001599
00001599
00001599
00001599
00001599
00001599

```

```

01600 T 0356
01601 T 0356
01602 T 0357

```

```

00001602
00001602
00001602

```

```

01603 T 0366
01604 T 0367
01605 T 0367
01606 T 0369
01607 T 0371

```

```

00001605
00001607
00001607
00001607
00001607
00001607
00001607
00001607
00001607

```

```

01608 T 0433

```

```

01609 T 0433

```

```

00001601
00001609
00001609
00001609
00001609
00001609
00001609
00001609
00001609
00001609
00001609
00001609
00001609
00001609
00001609

```

```

01610 T 0461

```

```

00001610
00001610

```

```

01611 T 0468

```

```

00001592

```

```

01612 T 0469

```

```

01613 T 0470

```

```

01614 T 0472

```

```

01615 T 0473

```

```

00001613

```

```

01616 T 0476

```

```

00001616
00001616

```

```

01617 T 0485

```

```

CURTYPE:=TYPEINDEX;%
CURMODE:=BITPATTERN;%
END;%

END OF SET CONSTANT ELSE BEGIN ERROR(99); INSYMBOL END;%

END OF FACTOR;%

%
%
PROCEDURE TERM;
BEGIN
INTEGER STARTSYM,MODE,TYPE1,MULOPTR,F;%
PUTDUMMY; STARTSYM:=NUMSYMS;%

FACTOR;%
MODE:=CURMODE;%
WHILE CURSY≧ASTERISK AND CURSY≠MODSY DO % "*" ,"/" ,"DIV" ,"MOD" ,"AND"
BEGIN%
TYPE1:=CURTYPE; MULOPTR:=CURSY;%
F:=TYPETAB1[TYPE1],FORM;%
IF F=NUMERIC OR F=FLOATING THEN%
BEGIN%
MODE:=NUMBER;%
IF CURSY=ASTERISK THEN PUTSYM("x") ELSE%

IF CURSY=SLASH THEN PUTSYM("/") ELSE%

IF CURSY=ANDSY THEN ERROR(64) ELSE%
BEGIN%
IF F=FLOATING THEN ERROR(64);%
IF CURSY=DIVSY THEN PUTTEXT(" DIV") ELSE PUTTEXT(" MOD");%

END END ELSE%

IF CURTYPE=BOOLTYPE OR F=SET THEN%
BEGIN%
MODE:=BITPATTERN;%
IF CURMODE≠MODE THEN%
BEGIN SYMTAB[STARTSYM]:=" B("; PUTSYM(")") END;%

```

```

01618 T 0488
01619 T 0489
01620 T 0490
00001590
01621 T 0490
00001588
00001621
01622 T 0492
00001371
12 IS 498 LONG, NEXT SEG 2

01623 T 0375
01624 T 0375
01625 T 0375
01626 T 0375
01627 T 0375
START OF SEGMENT ***** 14
01628 T 0000
00001628
00001628
00001628

01629 T 0006
01630 T 0007
01631 T 0007
01632 T 0010
01633 T 0010
01634 T 0011
01635 T 0013
01636 T 0015
01637 T 0015
01638 T 0016
00001638
00001638
00001638

01639 T 0026
00001639
00001639
00001639

01640 T 0035
01641 T 0038
01642 T 0038
01643 T 0040
00001643
00001643
00001643
00001643

01644 T 0055
00001641
00001636

01645 T 0055
01646 T 0059
01647 T 0060
01648 T 0061
01649 T 0061

```

```

PUTTEXT(" AND ");%

IF CURSY%(IF F=SET THEN ASTERISK ELSE ANDSY) THEN ERROR(64);%
END ELSE ERROR(64);%

PUTDUMMY; STARTSYM:=NUMSYMS;%

INSYMBOL; FACTOR;%
IF CURTYPE>0 AND TYPE1>0 THEN%
BEGIN%
IF CURTYPE≠TYPE1 THEN%
BEGIN%
IF TYPETAB1[TYPE1].FORM≠NUMERIC OR CURTYPE≠REALTYPE THEN%
CHECKTYPES(TYPE1,CURTYPE);%

IF TYPE1=REALTYPE THEN CURTYPE:=REALTYPE;%
END;%

IF CURTYPE=REALTYPE AND MULOPTR≥DIVSY THEN ERROR(65);%
END;%

IF MULOPTR=SLASH THEN CURTYPE:=REALTYPE;%
IF CURTYPE=0 THEN CURTYPE:=TYPE1;%
END OF WHILE LOOP;%

IF MODE=BITPATTERN AND CURMODE≠MODE THEN%
BEGIN SYMTAB[STARTSYM]:=" B("; PUTSYM(")") END;%

CURMODE:=MODE;%
END OF TERM;%

%
%
PROCEDURE SIMPLEEXPRESSION;
BEGIN
INTEGER STARTSYM,MODE,TYPE1,F;%

```

```

*** SIMPLE EXPRESSION ***
*****

```

```

00001649
00001649
00001649
00001649
01650 T 0072 00001650
00001650
01651 T 0078
01652 T 0082 00001648
01653 T 0084 00001653
00001653
00001653
01654 T 0091
01655 T 0092
01656 T 0094
01657 T 0094
01658 T 0095
01659 T 0095
01660 T 0098 00001660
00001660
00001660
00001660
00001660
00001660
00001660
00001660
01661 T 0159
01662 T 0161 00001658
01663 T 0161
01664 T 0164 00001656
01665 T 0164
01666 T 0166
01667 T 0168 00001632
01668 T 0168
01669 T 0170 00001669
00001669
00001669
00001669
01670 T 0180
01671 T 0181 00001626
14 IS 185 LONG, NEXT SEG 2
01672 T 0375
01673 T 0375
01674 T 0375
01675 T 0375
01676 T 0375

```

	START OF SEGMENT	*****	15
BOOLEAN SIGNED; %	01677	T	0000
%	01678	T	0000
PUTDUMMY; STARTSYM:=NUMSYMS; %	01679	T	0000
			00001679
			00001679
			00001679
IF CURSY=PLUS OR CURSY=MINUS THEN %	01680	T	0000
BEGIN SIGNED:=TRUE; %	01681	T	0008
PUTSYM(IF CURSY=PLUS THEN "+" ELSE "="); %	01682	T	0009
			00001682
			00001682
			00001682
INSYMBOL; %	01683	T	0020
END; %	01684	T	0021
			00001681
TERM; %	01685	T	0021
MODE:=CURMODE; %	01686	T	0021
IF SIGNED THEN %	01687	T	0022
BEGIN F:=TYPETAB1[CURTYPE],FORM; %	01688	T	0022
IF F≠NUMERIC AND F≠FLOATING THEN ERROR(29); %	01689	T	0024
END; %	01690	T	0029
			00001688
WHILE CURSY≥PLUS AND CURSYSORSY DO % "+" , "-" , "OR"	01691	T	0027
BEGIN %	01692	T	0030
TYPE1:=CURTYPE; F:=TYPETAB1[TYPE1],FORM; %	01693	T	0030
IF F=NUMERIC OR F=FLOATING THEN %	01694	T	0032
BEGIN MODE:=NUMBER; %	01695	T	0034
IF CURSY=PLUS THEN PUTSYM("+") ELSE %	01696	T	0035
			00001696
			00001696
			00001696
IF CURSY=MINUS THEN PUTSYM("=") ELSE ERROR(64); %	01697	T	0044
			00001697
			00001697
			00001697
END ELSE %	01698	T	0054
			00001695
IF CURTYPE=BOOLTYPE THEN %	01699	T	0054
BEGIN %	01700	T	0056
MODE:=BITPATTERN; %	01701	T	0056
IF CURMODE≠MODE THEN %	01702	T	0057
BEGIN SYMTAB[STARTSYM]:=" B(") PUTSYM(") " END; %	01703	T	0058
			00001703
			00001703
			00001703
			00001703
IF CURSY=ORSY THEN PUTTEXT(" OR") ELSE ERROR(64); %	01704	T	0068
			00001704
			00001704
END ELSE %	01705	T	0077
			00001700
IF F=SET THEN %	01706	T	0077
BEGIN %	01707	T	0079
MODE:=BITPATTERN; %	01708	T	0079
IF CURMODE≠MODE THEN %	01709	T	0080
BEGIN SYMTAB[STARTSYM]:=" B(") PUTSYM(") " END; %	01710	T	0081

```

                                00001710
                                00001710
                                00001710
                                00001710
IF CURSY=PLUS THEN PUTTEXT(" OR") ELSE%
                                01711 T 0091 00001711
                                00001711
IF CURSY=MINUS THEN BEGIN PUTTEXT(" AND");PUTTEXT(" NOT ")END
                                01712 T 0098 00001712
                                00001712
                                00001712
                                00001712
ELSE ERROR(64);%
                                01713 T 0114 00001712
END ELSE ERROR(64);%
                                01714 T 0118 00001707
INSYMBOL;%
                                01715 T 0118
PUTDUMMY; STARTSYM:=NUMSYMS;%
                                01716 T 0118
                                00001716
                                00001716
                                00001716
TERM;%
                                01717 T 0125
IF CURTYPE>0 AND TYPE1>0 THEN%
                                01718 T 0125
BEGIN%
                                01719 T 0127
IF CURTYPE≠TYPE1 THEN%
                                01720 T 0127
BEGIN%
                                01721 T 0128
IF TYPETAB1[TYPE1].FORM≠NUMERIC OR CURTYPE≠REALTYPE THEN%
                                01722 T 0129
CHECKTYPES(TYPE1,CURTYPE);%
                                01723 T 0131
                                00001723
                                00001723
                                00001723
                                00001723
                                00001723
                                00001723
                                00001723
IF TYPE1=REALTYPE THEN CURTYPE:=REALTYPE;%
                                01724 T 0192
END END;%
                                01725 T 0194
                                00001721
                                00001719
IF CURTYPE=0 THEN CURTYPE:=TYPE1;%
                                01726 T 0194
END OF WHILE LOOP;%
                                01727 T 0196
                                00001692
IF MODE=BITPATTERN AND CURMODE≠BITPATTERN THEN%
                                01728 T 0196
BEGIN SYMTAB[STARTSYM]:=" B("; PUTSYM(")") END;%
                                01729 T 0198
                                00001729
                                00001729
                                00001729
                                00001729
CURMODE:=MODE;%
                                01730 T 0208
END OF SIMPLEEXPRESSION;%
                                01731 T 0209
                                00001675
15 IS 213 LONG, NEXT SEG 2
                                01732 T 0375

```

x

```

%
PROCEDURE EXPRESSION;
BEGIN
  INTEGER STARTSYM, FIRSTSYM, TYPE1, RELOPTR, F;
  %*** EXPRESSION ***
  %*****

  BOOLEAN CALLGEN;
  %
  EXPRLEVEL:=EXPRLEVEL+1;
  IF EXPRLEVEL = 1 THEN
  BEGIN
    PUTDUMMY;

    FIRSTSYM := NUMSYMS;
  END;

  PUTDUMMY; STARTSYM:=NUMSYMS;

  PUTDUMMY;

  SIMPLEXPRESSION;
  IF CURSY>LSSSY AND CURSY<INSY THEN % "<","≤","≥",">","=","≠","IN"
  BEGIN
    TYPE1:=CURTYPE; F:=TYPETAB1[TYPE1],FORM;
    RELOPTR:=CURSY;
    IF F<ALFA THEN
    BEGIN
      IF CURMODE=BITPATTERN THEN
      BEGIN SYMTAB[STARTSYM]:=" REAL("; PUTSYM(") ) END;

      IF CURSY=LSSSY THEN PUTSYM("<") ELSE
      IF CURSY=LEQSY THEN PUTSYM("≤") ELSE
      IF CURSY=GEQSY THEN PUTSYM("≥") ELSE
      IF CURSY=GTRSY THEN PUTSYM(">") ELSE
      IF CURSY=EQLSY THEN PUTSYM("=") ELSE

```

```

01733 T 0375
01734 T 0375
01735 T 0375
01736 T 0375
START OF SEGMENT ***** 16
01737 T 0000
01738 T 0000
01739 T 0000
01740 T 0001
01741 T 0002
01742 T 0002
00001742
00001742
00001742
01743 T 0008
01744 T 0009
00001741
01745 T 0000
00001745
00001745
00001745
01746 T 0017
00001746
00001746
00001746
01747 T 0024
01748 T 0025
01749 T 0026
01750 T 0027
01751 T 0029
01752 T 0030
01753 T 0031
01754 T 0031
01755 T 0032
00001755
00001755
00001755
00001755
01756 T 0043
00001756
00001756
00001756
01757 T 0052
00001757
00001757
00001757
01758 T 0061
00001758
00001758
00001758
01759 T 0070
00001759
00001759
00001759
01760 T 0080
00001760
00001760

```

IF CURSY=NEQSY THEN PUTSYM("#") ELSE%	01761 T	0089	00001760
			00001761
			00001761
BEGIN%	01762 T	0098	00001761
IF F>FLOATING THEN ERROR(64));%	01763 T	0099	
SYMTAB[STARTSYM]:=INTST("); PUTSYM(","); CALLGEN:=TRUE;%	01764 T	0101	
			00001764
			00001764
			00001764
END;%	01765 T	0112	
END ELSE%	01766 T	0112	00001762
			00001753
IF F=SET THEN%	01767 T	0112	
BEGIN%	01768 T	0113	
IF CURMODE=BITPATTERN THEN%	01769 T	0114	
BEGIN SYMTAB[STARTSYM+1]:=REAL("); PUTSYM(")") END;%	01770 T	0114	
			00001770
			00001770
			00001770
			00001770
IF CURSY=EQLSY OR CURSY=NEQSY THEN%	01771 T	0125	
BEGIN PUTSYM(IF CURSY=EQLSY THEN "=" ELSE "#");%	01772 T	0127	
			00001772
			00001772
			00001772
END ELSE%	01773 T	0137	
			00001772
BEGIN%	01774 T	0137	
IF CURSY=LEQSY THEN SYMTAB[STARTSYM]:=INCL1(" ELSE%	01775 T	0137	
IF CURSY=GEQSY THEN SYMTAB[STARTSYM]:=INCL2(" ELSE ERROR(64);	01776 T	0140	
PUTSYM(","); CALLGEN:=TRUE;%	01777 T	0146	
			00001777
			00001777
			00001777
END END ELSE%	01778 T	0155	
			00001774
			00001768
IF F=POINTERS THEN%	01779 T	0155	
BEGIN%	01780 T	0156	
IF CURSY=EQLSY THEN PUTSYM("=") ELSE%	01781 T	0156	
			00001781
			00001781
			00001781
IF CURSY=NEQSY THEN PUTSYM("#") ELSE ERROR(64));%	01782 T	0165	
			00001782
			00001782
			00001782
END ELSE ERROR(64));%	01783 T	0176	
			00001780
INSYMBOL;%	01784 T	0177	
PUTDUMMY; STARTSYM:=NUMSYMS;%	01785 T	0177	
			00001785
			00001785
			00001785

```

SIMPLEEXPRESSION;%
IF CURTYPE>0 AND TYPE1>0 THEN%
  IF CURTYPE#TYPE1 THEN%
    IF RELOPTR#INSY THEN%
      BEGIN%
        IF TYPETAB1[TYPE1].FORM#NUMERIC OR CURTYPE#REALTYPE THEN%
          CHECKTYPES(TYPE1,CURTYPE);%

```

```

01786 T 0184
01787 T 0184
01788 T 0186
01789 T 0187
01790 T 0189
01791 T 0189
01792 T 0192

```

```

00001792
00001792
00001792
00001792
00001792
00001792
00001792

```

```

END ELSE%

```

```

01793 T 0253

```

```

  IF TYPETAB1[CURTYPE].FORM#SET THEN ERROR(66)%
  ELSE CHECKTYPES(TYPE1,TYPETAB1[CURTYPE],SETTYPE);%

```

```

01794 T 0253
01795 T 0256

```

```

00001790
00001795
00001795
00001795
00001795
00001795
00001795
00001795

```

```

IF CURMODE#BITPATTERN THEN%
BEGIN SYMTAB[STARTSYM] := " REAL("); PUTSYM(")"); END;%

```

```

01796 T 0318
01797 T 0318

```

```

00001797
00001797
00001797
00001797

```

```

IF CALLGEN THEN PUTSYM(")");%

```

```

01798 T 0329

```

```

00001798
00001798
00001798

```

```

CURTYPE:=BOOLTYPE; CURMODE:=BITPATTERN;%
END;%

```

```

01799 T 0337
01800 T 0339

```

```

00001749

```

```

EXPRLEVEL:=EXPRLEVEL-1;%
IF EXPRLEVEL=0 THEN%
BEGIN%

```

```

01801 T 0339
01802 T 0341
01803 T 0342
01804 T 0342
01805 T 0343
01806 T 0343
01807 T 0345

```

```

  IF CURMODE#BITPATTERN THEN%
  BEGIN%
    SYMTAB[FIRSTSYM] := " REAL(";%
    PUTSYM(")");%

```

```

00001807
00001807
00001807

```

```

END;%

```

```

01808 T 0353

```

```

00001808

```

```

WRITEEXPR;%

```

```

01809 T 0353

```

```

00001809
00001809
00001809
00001809
00001809
00001809

```


END;%
END OF EXPRESSION;%

00001809
01810 T 0432
00001803
01811 T 0432
00001735
16 IS 439 LONG, NEXT SEG 2

%
%
DEFINE BOOLEXP=#
BEGIN%
 PUTDUMMY; EXPRLEVEL:=1; EXPRESSION;%
 IF CURTYPE>0 THEN IF CURTYPE#BOOLTYPE THEN ERROR(17);%
 IF CURMODE#BITPATTERN THEN%
 BEGIN SYMTAB[1]#=" B("; PUTSYM(")") END;%
 EXPRLEVEL:=0; WRITEEXPR;%
END OF BOOLEXP#;%

01812 T 0375
01813 T 0375
01814 T 0375
01815 T 0375
01816 T 0375
01817 T 0375
01818 T 0375
01819 T 0375
01820 T 0375
01821 T 0375

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X
X
X
X          PART 5:  INTRINSIC ROUTINES,
X          -----
X
X
X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X
X
PROCEDURE CONCAT;          **** "CONCAT" ****
BEGIN                      *****
    DEFINE INEXPR=%
        BEGIN INSYMBOL; EXPRESSION;%
            IF CURTYPE>0 THEN%
                IF TYPETAB1[CURTYPE].FORM#NUMERIC THEN ERROR(17));%
        END #;%
X
    PUTTEXT("CONCAT"); PUTSYM("(");%

    INSYMBOL;%
    IF CURSY=LPAR THEN%
        BEGIN%
            INSYMBOL; EXPRESSION;%
            IF CURTYPE>0 THEN%
                IF TYPETAB1[CURTYPE].FORM>ALFA THEN ERROR(17));%
            IF CURSY=COMMA THEN%
                BEGIN%
                    PUTSYM(","); INSYMBOL; EXPRESSION;%

                    IF CURTYPE>0 THEN%
                        IF TYPETAB1[CURTYPE].FORM>ALFA THEN ERROR(17));%
                    IF CURSY=COMMA THEN%
                        BEGIN%
                            PUTSYM(","); INEXPR;%

                            IF CURSY=COMMA THEN%
                                BEGIN%
                                    PUTSYM(","); INEXPR;%

                                    IF CURSY=COMMA THEN%
                                        BEGIN%

```

```

01823 T 0375
01824 T 0375
01825 T 0375
01826 T 0375
01827 T 0375
01828 T 0375
01829 T 0375
01830 T 0375
01831 T 0375
01832 T 0375
01833 T 0375
01834 T 0375
01835 T 0375
01836 T 0375
01837 T 0375
START OF SEGMENT ***** 17
01838 T 0000
01839 T 0000
01840 T 0000
01841 T 0000
01842 T 0000
01843 T 0000
00001843
00001843
00001843
00001843
00001843
01844 T 0014
01845 T 0015
01846 T 0015
01847 T 0016
01848 T 0017
01849 T 0018
01850 T 0021
01851 T 0022
01852 T 0022
00001852
00001852
00001852
01853 T 0031
01854 T 0031
01855 T 0035
01856 T 0035
01857 T 0036
00001857
00001857
00001857
00001857
01858 T 0048
01859 T 0049
01860 T 0050
00001860
00001860
00001860
00001860
01861 T 0062
01862 T 0063

```

```

PUTSYM(","); INEXPR;%

PUTSYM(","); PUTCONST(CARDCNT);%

PUTSYM(")");%

IF CURSY≠RPAR THEN BEGIN ERROR(3); SKIP(RPAR) END;%
END ELSE BEGIN ERROR(3); SKIP(RPAR) END;%

END ELSE BEGIN ERROR(3); SKIP(RPAR) END;%

END ELSE BEGIN ERROR(3); SKIP(RPAR) END;%

END ELSE BEGIN ERROR(3); SKIP(RPAR) END;%

END ELSE BEGIN ERROR(3); SKIP(RPAR) END;%

CURTYPE:=REALTYPE;%
IF CURSY=RPAR THEN INSYMBOL;%
END OF CONCAT;%

```

```

%
%
PROCEDURE PREAD(CHANGELINE);%
VALUE CHANGELINE; BOOLEAN CHANGELINE;%
BEGIN%
  INTEGER FILEID,F;%

  BOOLEAN CHECK;%
  GEN(" BEGIN",7,2);%

  FILEPARAM(INPUTFILE); FILEID:=FILENAME;%

```

```

01863 T 0063
00001863
00001863
00001863
00001863

01864 T 0076
00001864
00001864
00001864
00001864
00001864
00001864
00001864
00001864

01865 T 0096
00001865
00001865
00001865

01866 T 0104
00001866

01867 T 0106
00001862
00001867

01868 T 0108
00001859
00001868

01869 T 0110
00001856
00001869

01870 T 0112
00001851
00001870

01871 T 0114
00001846
00001871

01872 T 0116
01873 T 0117
01874 T 0119
00001836

17 IS 120 LONG, NEXT SEG 2

01875 T 0375
01876 T 0375
01877 T 0375
01878 T 0375
01879 T 0375
01880 T 0375
START OF SEGMENT ***** 18
01881 T 0000
01882 T 0000
00001882
01883 T 0009
00001888
00001883

```

```

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;%

```

```

01884 T 0062
01885 T 0065
01886 T 0066
01887 T 0066
01888 T 0067
01889 T 0068
01890 T 0070
01891 T 0071
01892 T 0071

```

```

00001883
00001883
00001883
00001883
00001883
00001883
00001883
00001883

```

```

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;%
      WRITEEXPR; GEN(":",2,6));%

```

```

00001892
00001892
00001892
00001892
01893 T 0100
01894 T 0100
01895 T 0100
01896 T 0102
01897 T 0104
01898 T 0105
01899 T 0109
01900 T 0109
01901 T 0110
01902 T 0112

```

```

00001902
00001902
00001902
00001902
00001902
00001902
00001902
00001902

```

```

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));%

```

```

01903 T 0199
01904 T 0209
01905 T 0239
01906 T 0260
01907 T 0281
01908 T 0329

```

```

00001903
00001904
00001904
00001904
00001905
00001905
00001906
00001906
00001907
00001907
00001908
00001908

```

```

GEN(" ",1,7); GENINT(CARDCNT); GEN(" ",1,7);%
01909 T 0418 00001908
00001908
00001909
00001909
00001909
00001909

IF CHECK THEN%
BEGIN%
GEN(" ",1,7); GENINT(TYPETAB2[CURTYPE]); GEN(" ",1,7);%
01910 T 0480
01911 T 0480
01912 T 0481
00001912
00001912
00001912
00001912

GENINT(TYPETAB3[CURTYPE]); GEN(" ",1,7);%
01913 T 0543
00001913
00001913
00001913

GENINT(CARDCNT); GEN(" ",1,7);%
01914 T 0596
00001914
00001914
00001914

END;%
01915 T 0649
END ELSE BEGIN ERROR(82); INSYMBOL END;%
01916 T 0649
00001900
00001916

END ELSE BEGIN ERROR(8); INSYMBOL END;%
01917 T 0652
00001897
00001917

END ELSE BEGIN ERROR(1); INSYMBOL END;%
01918 T 0655
00001894
00001918

END ELSE ERROR(9);%
01919 T 0658
00001891

GEN(" ",1,7);%
01920 T 0660
00001920

END UNTIL CURSY#COMMA;%
01921 T 0669
00001888

IF CURSY#RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;%
01922 T 0671
00001922

IF CURSY=RPAR THEN INSYMBOL;%
01923 T 0673
END;%
01924 T 0675
00001888

IF CHANGELINE THEN%
BEGIN%
GEN("RLINE(",6,2); GENID("F",FILEID,5); GEN(" ",1,7);%
01925 T 0675
01926 T 0676
01927 T 0676
00001927
00001927
00001927

GENID("V",FILEID,5); GEN(" ",1,7);%
01928 T 0706
00001928
00001928

GENID("I",FILEID,5); GEN(" ",1,7);%
01929 T 0727
00001929
00001929

END;%
01930 T 0748

```

```
GEN("END",4,5);%
END OF PREAD);%
```

```
00001926
01931 T 0748
00001931
01932 T 0757
00001879
18 IS 773 LONG, NEXT SEG 2
```

```
%
%
PROCEDURE PWRITE(LINEFEED);%
VALUE LINEFEED; BOOLEAN LINEFEED;%
BEGIN%
  INTEGER FILEID,F,I,LASTSY;%
  POINTER P;%
  GEN(" BEGIN",7,2);%
  FILEPARAM(OUTPUTFILE); FILEID:=FILENAME;%

```

```
01933 T 0375
01934 T 0375
01935 T 0375
01936 T 0375
01937 T 0375
01938 T 0375
START OF SEGMENT ***** 19
01939 T 0000
01940 T 0000
```

```
IF TYPETAB[CURTYPE],FORM=FILES THEN ERROR(85);%
IF SYMKIND[CURSY]≠TERMINAL THEN%
BEGIN%
  IF CURSY NEQ HPAR THEN%
  DO BEGIN%
    WHILE CURSY=COMMA DO INSYMBOL;%
    IF CURSY=ALFACONST AND CURLLENGTH>7 THEN%
    BEGIN%
      GEN("WALFA",6,2); GENID("F",FILEID,5); GEN(" ",1,7);%
```

```
01941 T 0009
00001940
00001941
00001941
00001941
00001941
00001941
00001941
00001941
00001941
00001941
00001941
00001941
01942 T 0062
01943 T 0065
01944 T 0066
01945 T 0066
01946 T 0067
01947 T 0068
01948 T 0070
01949 T 0072
01950 T 0072
```

```
GENID("V",FILEID,5); GEN(" ",1,7);%
GENID("I",FILEID,5); GEN(" ",1,7);%
```

```
00001950
00001950
00001950
01951 T 0102
00001951
00001951
01952 T 0123
00001952
00001952
```

```
P:=STRINGPNT;%
FOR I:=1 STEP 7 UNTIL 80 DO%
  IF I≤CURLLENGTH THEN%
  BEGIN%
    IF ALGOLCNT<10 THEN WRITEALGOL;%
    REPLACE ALGOLPNT;ALGOLPNT BY " ", PIP FOR 7, " ", " ", "%";
    ALGOLCNT:=ALGOLCNT+1;%
  END ELSE GEN("O",2,6);%
```

```
01953 T 0144
01954 T 0145
01955 T 0147
01956 T 0147
01957 T 0148
01958 T 0150
01959 T 0161
01960 T 0163
```

```

GENINT(CURLNGTH); GEN(" ",1,7);%

GENINT(CARDCNT); GEN(" ",1,7);%

INSYMBOL;%
END OF ALFACONST ELSE%
BEGIN%
GEN("PWRITE(",7,1); GENID("F",FILEID,5); GEN(" ",1,7);%

GENID("V",FILEID,5); GEN(" ",1,7);%

GENID("I",FILEID,5); GEN(" ",1,7);%

LASTSY:=CURSY;%
EXPRESSION; F:=TYPETAB1[CURTYPE],FORM;%
GEN(" ",1,7);%

IF F=NUMERIC OR F=FLOATING OR F=CHAR OR F=ALFA OR%
CURTYPE=BOOLTYPE THEN%
BEGIN%
IF F=NUMERIC THEN GENINT(1) ELSE%

IF F=FLOATING THEN GENINT(2) ELSE%

IF F=ALFA THEN GENINT(5) ELSE%

IF F=CHAR THEN GENINT(4) ELSE GENINT(3);%

GEN(" ",1,7);%

IF CURSY=COLON THEN%
BEGIN%
INSYMBOL; EXPRESSION;%
IF TYPETAB1[CURTYPE].FORM NEQ NUMERIC THEN ERROR(17);%
GEN(" ",1,7);%

IF CURSY=COLON THEN%
BEGIN%
IF F=FLOATING THEN ERROR(4);%
INSYMBOL; EXPRESSION;%

```

```

00001956
00001960
01961 T 0174 00001961
00001961
00001961
01962 T 0227 00001962
00001962
00001962
01963 T 0280
01964 T 0280 00001949
01965 T 0280
01966 T 0281 00001966
00001966
00001966
01967 T 0311 00001967
00001967
01968 T 0332 00001968
00001968
01969 T 0353
01970 T 0354
01971 T 0356 00001971
01972 T 0369
01973 T 0368
01974 T 0369
01975 T 0370 00001975
00001975
01976 T 0416 00001976
00001976
01977 T 0461 00001977
00001977
01978 T 0506 00001978
00001978
00001978
00001978
01979 T 0595 00001979
01980 T 0604
01981 T 0605
01982 T 0605
01983 T 0606
01984 T 0609 00001984
01985 T 0618
01986 T 0619
01987 T 0619
01988 T 0621

```

```

        IF TYPETAB1[CURTYPE].FORM NEQ NUMERIC THEN ERROR(17));%
        GEN(" ",1,7));%
    END ELSE GEN(" ",3,5));%

END ELSE%

BEGIN%
    IF F=FLOATING THEN GENINT(16) ELSE%

        IF F=ALFA AND LASTSY=ALFACONST THEN GENINT(CURLNGTH) ELSE

            IF F=ALFA THEN GENINT(7) ELSE%

                IF F=CHAR THEN GENINT(1) ELSE GENINT(10));%

                    GEN(" ",4,4));%

            END;%

        END ELSE ERROR(17));%

        GENINT(CARDCNT); GEN(" ",1,7));%

    END OF EXPRESSION;%

    GEN("; ",1,7));%

    END UNTIL CURSY=COMMA;%

    IF CURSY=RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;%

    IF CURSY=RPAR THEN INSYMBOL;%

END;%

FILENAME=FILEID;%
IF LINEFEED THEN%
BEGIN%
    INTEGER DUMMY;%

    GEN("WLINE(",6,2); GENID("F",FILENAME,5); GEN(" ",1,7));%

    GENID("V",FILENAME,5); GEN(" ",1,7));%

    GENID("I",FILENAME,5); GEN(" ",1,7));%

```

```

01989 T 0622
01990 T 0625
01991 T 0634 00001990
01992 T 0643 00001985
01993 T 0643 00001991
01994 T 0646 00001981
01995 T 0691 00001994
01996 T 0737 00001994
01997 T 0782 00001995
01998 T 0871 00001995
01999 T 0880 00001996
02000 T 0880 00001996
02001 T 0882 00001997
02002 T 0935 00001997
02003 T 0935 00001997
02004 T 0945 00001997
02005 T 0946 00001997
02006 T 0949 00001998
02007 T 0951 00001998
02008 T 0951 00001993
02009 T 0952 00001974
02010 T 0953 00002001
02011 T 0953 00002001
02012 T 0000 00002001
02013 T 0030 00001965
02014 T 0051 00002003
00001946
00002005
00001944
00002012
00002012
00002012
00002019
00002019

```

START OF SEGMENT ***** 20


```

END; %
GEN("END",4,5); %
END OF PWRITE; %

```

```

%
%
PROCEDURE FILEHANDLING(PROCNUM);
VALUE PROCNUM; INTEGER PROCNUM;
BEGIN
  INTEGER F;

  CASE PROCNUM OF
  BEGIN ;
    GEN("PUT",3,5);

    GEN("GET",3,5); %
    GEN("RESET",5,3); %
    GEN("REWRITE",7,1); %
    GEN("PAGE",4,4); %
  END; %

  GEN("(",1,7); FILEPARAM(0); %

```

```

%*** FILE HANDLING PROCEDURES:
%***
%*** 1) PUT
%*** 2) GET
%*** 3) RESET
%*** 4) REWRITE
%*** 5) PAGE

```

```

IF FILENAME=0 THEN ERROR(78); %
F:=TYPETAB1[CURTYPE],FORM; %
IF F=FILES AND PROCNUM=5 THEN ERROR(80); %
GENID("F",FILENAME,5); GEN("(",1,7); %

GENID("V",FILENAME,5); GEN("(",1,7); %

```

```

00002014
00002014
02015 T 0072 00002010
20 IS 74 LONG, NEXT SEG 19
02016 T 0954 00002016
02017 T 0963 00001937
19 IS 975 LONG, NEXT SEG 2
02018 T 0375
02019 T 0375
02020 T 0375
02021 T 0375
02022 T 0375
02023 T 0375
START OF SEGMENT ***** 21
02024 T 0000
02025 T 0000
02026 T 0000 00002026
02027 T 0010 00002027
02028 T 0019 00002028
02029 T 0029 00002029
02030 T 0038 00002030
02031 T 0048 00002025
START OF SEGMENT ***** 22
22 IS 6 LONG, NEXT SEG 21
02032 T 0047 00002032
00002032
00002032
00002032
00002032
00002032
00002032
00002032
00002032
00002032
00002032
02033 T 0113
02034 T 0115
02035 T 0117
02036 T 0120 00002036
00002036
02037 T 0141 00002037
00002037

```

```

GENID("I",FILENAME,5); GEN(", ",1,7);%

GENINT(CARDCNT); GEN(", ",1,7);%

IF CURSY≠RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;%

IF CURSY=RPAR THEN INSYMBOL;%
END OF FILEHANDLING;%

%
%
PROCEDURE PACK;%
BEGIN%
  INTEGER IT,T;%

  GEN("PACK(",5,3);%

  INSYMBOL;%
  IF CURSY=LPAR THEN%
  BEGIN%
    INSYMBOL;%
    IF CURSY=IDENTIFIER THEN%
    BEGIN%
      SEARCH;%

      IF FOUND THEN%
      BEGIN%
        IF THISID.IDCLASS=VAR THEN%
        BEGIN%
          T:=TYPETAB1[THISID,TYPE];%
          IF T.FORM=ARRAYS THEN%
          BEGIN%
            IT:=T.INXTYPE;%
            IF TYPETAB1[T,ARRTYPE],FORM≠CHAR THEN ERROR(88);%
            GENID("V",1000×THISLEVEL+THISINDEX,5);%

            IF THISLEVEL>1 AND THISLEVEL≠CURLEVEL THEN ERROR(5);%
            GEN(", ",1,7); GENINT(TYPETAB2[THISID,TYPE]);%

            GEN(", ",1,7); GENINT(TYPETAB3[THISID,TYPE]);%
          END%
        END%
      END%
    END%
  END%

```

```

00002037
02038 T 0162
00002038
00002038
02039 T 0183
00002039
00002039
00002039
02040 T 0235
00002040
02041 T 0238
02042 T 0239
00002022
21 IS 242 LONG, NEXT SEG 2

02043 T 0375
02044 T 0375
02045 T 0375
02046 T 0375
02047 T 0375
START OF SEGMENT ***** 23
02048 T 0000
00002048
02049 T 0009
02050 T 0009
02051 T 0010
02052 T 0010
02053 T 0011
02054 T 0012
02055 T 0012
00002055
00002055
00002055
00002055
02056 T 0042
02057 T 0043
02058 T 0043
02059 T 0044
02060 T 0045
02061 T 0046
02062 T 0048
02063 T 0048
02064 T 0049
02065 T 0053
00002065
02066 T 0066
02067 T 0069
00002067
00002067
00002067
02068 T 0123
00002068
00002068
00002068

```

```

        END ELSE ERROR(88));%
    END ELSE ERROR(88));%
    END ELSE ERROR(1));%
END ELSE ERROR(9));%

INSYMBOL;%
IF CURSY=COMMA THEN%
BEGIN%
    GEN(",","1,7");%

    INSYMBOL; EXPRESSION; CHECKTYPES(IT,CURTYPE);%

    IF CURSY=COMMA THEN%
    BEGIN%
        GEN(",","1,7");%

        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; WRITEEXPR;%

                    IF CURTYPE>0 THEN%
                    IF TYPETAB1[CURTYPE].FORM≠ALFA THEN ERROR(12);%
                    END ELSE ERROR(8));%

                END ELSE ERROR(1));%
            END ELSE ERROR(9));%
        END ELSE BEGIN ERROR(89); SKIP(RPAR) END;%
    
```

```

02069 T 0177
00002062
02070 T 0178
00002059
02071 T 0179
00002057
02072 T 0181
00002054
02073 T 0182
02074 T 0182
02075 T 0183
02076 T 0184
00002076
02077 T 0193
00002077
00002077
00002077
00002077
00002077
00002077
00002077
02078 T 0253
02079 T 0253
02080 T 0254
00002080
02081 T 0263
02082 T 0263
02083 T 0264
02084 T 0265
00002084
00002084
00002084
00002084
02085 T 0294
02086 T 0294
02087 T 0294
02088 T 0296
02089 T 0298
02090 T 0298
00002090
00002090
00002090
00002090
00002090
00002090
00002090
02091 T 0378
02092 T 0379
02093 T 0382
00002089
02094 T 0383
00002086
02095 T 0385
00002083
02096 T 0386
00002079
00002096

```

```

END ELSE BEGIN ERROR(89); SKIP(RPAR) END;%

IF CURSY#RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;%

IF CURSY=RPAR THEN INSYMBOL;%
END ELSE BEGIN ERROR(3); INSYMBOL END;%

GEN(", ", 1, 7); GENINT(CARDCNT); GEN(", ", 1, 7);%

END OF PACK;%

%
%
PROCEDURE UNPACK;%
BEGIN%
  INTEGER IT, T;%

  GEN("UNPACK(", 7, 1); INSYMBOL;%

  IF CURSY=LPAR THEN%
  BEGIN%
    INSYMBOL; EXPRESSION;%
    IF CURTYPE>0 THEN IF TYPETAB1[CURTYPE].FORM#ALFA THEN ERROR(17);%
    IF CURSY=COMMA THEN%
    BEGIN%
      GEN(", ", 1, 7); INSYMBOL;%

      IF CURSY=IDENTIFIER THEN%
      BEGIN%
        SEARCH;%

        IF FOUND THEN%
        BEGIN%
          IF THISID.IDCLASS=VAR THEN%
          BEGIN%
            T:=TYPETAB1[THISID.TYPE];%
            IF T.FORM=ARRAYS THEN%
            BEGIN%
              IT:=T, INXTYPE;%
              IF TYPETAB1[T,ARRTYPE].FORM#CHAR THEN ERROR(88);%
              IF THISLEVEL>1 AND THISLEVEL#CURLEVEL THEN ERROR(5);%
              GENID("V", 1000*THISLEVEL+THISINDEX, 5);%

              GEN(", ", 1, 7); GENINT(TYPETAB2[THISID.TYPE]);%

```

```

02097 T 0388
00002075
00002097
02098 T 0390
00002098
02099 T 0393
02100 T 0394
00002051
00002100
02101 T 0397
00002101
00002101
00002101
00002101
02102 T 0459
00002046
23 IS 465 LONG, NEXT SEG 2

02103 T 0375
02104 T 0375
02105 T 0375
02106 T 0375
02107 T 0375
START OF SEGMENT ***** 24
02108 T 0000
00002108
02109 T 0009
02110 T 0010
02111 T 0010
02112 T 0011
02113 T 0015
02114 T 0016
02115 T 0017
00002115
02116 T 0026
02117 T 0027
02118 T 0027
00002118
00002118
00002118
00002118
02119 T 0057
02120 T 0058
02121 T 0058
02122 T 0059
02123 T 0060
02124 T 0061
02125 T 0063
02126 T 0063
02127 T 0064
02128 T 0068
02129 T 0071
00002128
02130 T 0084

```

```

GEN(",",1,7); GENINT(TYPETAB3[THISID,TYPE]);%

END ELSE ERROR(88);%

END ELSE ERROR(88);%

END ELSE ERROR(1);%

END ELSE ERROR(9);%

INSYMBOL;%
IF CURSY=COMMA THEN%
BEGIN%
GEN(",",1,7);%

INSYMBOL; EXPRESSION; CHECKTYPES(IT,CURTYPE);%

END ELSE BEGIN ERROR(89); SKIP(RPAR) END;%

END ELSE BEGIN ERROR(89); SKIP(RPAR) END;%

IF CURSY#RPAR THEN BEGIN ERROR(89); SKIP(RPAR) END;%

IF CURSY=RPAR THEN INSYMBOL;%
END ELSE BEGIN ERROR(3); INSYMBOL END;%

GEN(",",1,7); GENINT(CARDCNT); GEN(",",1,7);%

END OF UNPACK;%

%
%
PROCEDURE NEWDISP;
BEGIN%
INTEGER T1;%

```

*** "NEW", "DISPOSE"

```

00002130
00002130
00002130
02131 T 0138 00002131
00002131
00002131
02132 T 0192 00002125
02133 T 0193 00002122
02134 T 0194 00002120
02135 T 0196 00002117
02136 T 0197
02137 T 0197
02138 T 0198
02139 T 0199 00002139
02140 T 0208 00002140
00002140
00002140
00002140
00002140
00002140
00002140
02141 T 0268 00002138
00002141
02142 T 0270 00002114
00002142
02143 T 0272 00002143
02144 T 0274
02145 T 0276 00002110
00002145
02146 T 0278 00002146
00002146
00002146
00002146
02147 T 0340 00002106
24 IS 344 LONG, NEXT SEG 2

02148 T 0375
02149 T 0375
02150 T 0375
02151 T 0375
02152 T 0375

```

```

IF CURNAME1="3000NEW" THEN GEN("NEW(",4,4) ELSE%
BEGIN GEN("DISPOSE",7,1); GEN("(",1,7) END;%

```

```

INSYMBOL;%
IF CURSY=LPAR THEN%
BEGIN%
INSYMBOL;%
IF CURSY=IDENTIFIER THEN%
BEGIN%
SEARCH;%

```

```

IF FOUND THEN%
BEGIN%
VARIABLE;%
IF CURTYPE>0 THEN IF TYPETAB1[CURTYPE],FORM=POINTERS THEN%
BEGIN%
WRITEEXPR; GEN("(",1,7);%

```

```

T1:=TYPETAB1[CURTYPE],POINTTYPE;%
T1:=TYPETAB1[T1],SIZE;%
IF T1>1023 THEN ERROR(86);%
GENINT(T1); GEN(")",1,7);%

```

```

END ELSE ERROR(81);%

```

```

END ELSE BEGIN ERROR(1); INSYMBOL END;%

```

```

END ELSE ERROR(9);%

```

```

WHILE CURSY=COMMA DO%
BEGIN INSYMBOL;%
IF CURSY NEQ IDENTIFIER THEN ERROR(9);%
IF CURSY NEQ RPAR THEN INSYMBOL;%
END;%

```

```

END ELSE BEGIN ERROR(58); SKIP(RPAR) END;%

```

```

IF CURSY#RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;%

```

```

START OF SEGMENT ***** 25
02153 T 0000
00002153
02154 T 0010
00002154
00002154
00002154
02155 T 0031
02156 T 0031
02157 T 0032
02158 T 0032
02159 T 0033
02160 T 0034
02161 T 0034
00002161
00002161
00002161
00002161
02162 T 0064
02163 T 0065
02164 T 0065
02165 T 0066
02166 T 0068
02167 T 0069
00002167
00002167
00002167
00002167
00002167
00002167
00002167
00002167
00002167
02168 T 0156
02169 T 0158
02170 T 0159
02171 T 0161
00002171
00002171
00002171
02172 T 0214
00002166
02173 T 0215
00002163
00002173
02174 T 0217
00002160
02175 T 0218
02176 T 0220
02177 T 0220
02178 T 0222
02179 T 0224
00002179
02180 T 0225
00002157
00002180
02181 T 0227
00002181

```

IF CURSY=RPAR THEN INSYMBOL;*

END OF NEWDISP;*

02182 T 0229

02183 T 0231

00002151

25 IS 234 LONG, NEXT SEG 2

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X
X
X
X
X
X
X
X
X
X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X
X
X
PROCEDURE STATEMENT; FORWARD;X
X
PROCEDURE ASSIGNMENT;X
BEGINX
  INTEGER LEFTTYPE;X
  LABEL ASSIGN,EXIT;X
  IF FOUND THENX
  BEGINX
    IF THISID.IDCLASS=VAR ORX
      THISID.IDCLASS=CONST AND BOOLEAN(THISID,FORMAL) THENX
    BEGINX
      VARIABLE; LEFTTYPE:=CURTYPE;X
    ASSIGN: IF CURSY≠ASSIGNSY THENX
      BEGIN ERROR(28); SKIP(ASSIGNSY);X
        IF SYMKIND[CURSY]=TERMINAL THEN GO TO EXIT;X
      END;X
    INSYMBOL;X
    IF TYPETAB1[LEFTTYPE].STRUCT>0 THENX
    BEGINX
      ERROR(95);X
    END ELSEX
    BEGINX
      WRITEEXPR; GEN("=",2,6);X
    IF CHECKOPTION AND TYPETAB1[LEFTTYPE].FORMSCHAR THENX
    CHECKEXPR(TYPETAB2[LEFTTYPE],TYPETAB3[LEFTTYPE]) ELSEX

```

```

02185 T 0375
02186 T 0375
02187 T 0375
02188 T 0375
02189 T 0375
02190 T 0375
02191 T 0375
02192 T 0375
02193 T 0375
02194 T 0375
02195 T 0375
02196 T 0375
02197 T 0375
02198 T 0375
02199 T 0375
02200 T 0375
02201 T 0375
02202 T 0375
START OF SEGMENT ***** 26
02203 T 0000
02204 T 0000
02205 T 0000
02206 T 0000
02207 T 0002
02208 T 0004
02209 T 0005
02210 T 0006
02211 T 0007
02212 T 0000
02213 T 0011
00002211
02214 T 0011
02215 T 0011
02216 T 0013
02217 T 0013
02218 T 0014
00002216
02219 T 0014
02220 T 0015
00002220
00002220
00002220
00002220
00002220
00002220
00002220
00002220
02221 T 0102
02222 T 0104
00002222
00002222
00002222
00002222
00002222
00002222
00002222
00002222

```



```

EXPRESSION;%
WRITEEXPR;%

CHECKTYPES(LEFTTYPE,CURTYPE);%

END;%
END ELSE%

BEGIN % FUNCTION ASSIGNMENT.
IF THISLEVEL#CURLEVEL=1 OR THISINDEX#CURFUNC THEN ERROR(5);%
GENID("V",1000*THISLEVEL+THISINDEX,5) LEFTTYPE:=THISID,TYPE;%

INSYMBOL; GO TO ASSIGN;%
END;%

END ELSE%

BEGIN%
SKIP(ASSIGNSY);%
IF CURSY=ASSIGNSY THEN GO TO ASSIGN;%
END;%

```

```

00002222
00002222
00002222
00002222
00002222
00002222
00002222
00002222
00002222
00002222
00002222
00002222
00002222
00002222
00002222
00002222
00002222
00002222
00002222
02223 T 0183
02224 T 0184
00002224
00002224
00002224
00002224
00002224
00002224
00002224
02225 T 0262
00002225
00002225
00002225
00002225
00002225
00002225
00002225
02226 T 0322
00002219
02227 T 0322
00002208
02228 T 0322
02229 T 0323
02230 T 0326
00002230
02231 T 0340
02232 T 0341
00002228
02233 T 0341
00002205
02234 T 0341
02235 T 0343
02236 T 0343
02237 T 0345
00002234

```

EXIT;%
END OF ASSIGNMENT;%

02238 T 0345
02239 T 0345
0000220;
26 IS 352 LONG, NEXT SEG 2

%
%
PROCEDURE COMPSTAT;%
BEGIN%
 INTEGER BEGINNUM;%

 LABEL STATM;%
%
 BEGINNUM:=NUMBEGINS:=NUMBEGINS+1; MARGIN(" B",BEGINNUM);%

 GEN("BEGIN",6,3);%

 DO BEGIN%
 IF CURSY=SEMICOLON OR CURSY=BEGINSY THEN INSYMBOL;%
STATM: STATEMENT;%
 GEN(")",1,7);%

 IF CURSY=ELSESY THEN BEGIN ERROR(20); INSYMBOL; GO STATM END;%
 IF SYMKIND[CURSY]=INITIAL THEN BEGIN ERROR(21); GO STATM END;%

 END UNTIL CURSY#SEMICOLON;%

 IF CURSY#ENDSY THEN%
 BEGIN ERROR(24); SKIP(ENDSY);%
 IF CURSY#ENDSY THEN BEGIN INSYMBOL; GO TO STATM END;%

 END;%

 GEN(" END",5,4); MARGIN(" E",BEGINNUM);%

 INSYMBOL;%
END OF COMPSTAT;%

02240 T 0375
02241 T 0375
02242 T 0375
02243 T 0375
02244 T 0375
START OF SEGMENT ***** 27
02245 T 0000
02246 T 0000
02247 T 0000
00002247
00002247
02248 T 0020
00002248
02249 T 0029
02250 T 0029
02251 T 0031
02252 T 0032
00002252
02253 T 0041
00002253
02254 T 0047
00002254
02255 T 0049
00002249
02256 T 0051
02257 T 0051
02258 T 0053
00002258
02259 T 0056
00002257
02260 T 0056
00002260
00002260
00002260
02261 T 0083
02262 T 0083
00002243
27 IS 88 LONG, NEXT SEG 2

%
%
PROCEDURE IFSTAT;%
BEGIN%
 LABEL EXIT;%

 GEN("IF",3,6);%

02263 T 0375
02264 T 0375
02265 T 0375
02266 T 0375
02267 T 0375
START OF SEGMENT ***** 28
02268 T 0000

INSYMBOL; BOOLEXP; %

02269 T 0009 00002268
00002269
00002269
00002269
00002269
00002269
00002269
00002269
00002269
00002269
00002269
00002269
00002269
00002269
00002269
00002269
00002269

IF CURSY#THENSY THEN%
BEGIN IF CURTYPE>0 THEN ERROR(27); %
SKIP(THENSY); %
IF CURSY#THENSY THEN%
BEGIN IF CURTYPE=0 THEN ERROR(27); %
IF SYMKIND(CURSY)=TERMINAL THEN GO TO EXIT; %
END; END; %

02270 T 0112
02271 T 0113
02272 T 0115
02273 T 0116
02274 T 0117
02275 T 0119
02276 T 0121

GEN(" THEN", 6, 3); %

00002274
00002271
02277 T 0121
00002277

INSYMBOL; STATEMENT; %
IF CURSY#ELSESY THEN%
BEGIN GEN(" ELSE", 6, 3); INSYMBOL; STATEMENT END; %

02278 T 0130
02279 T 0131
02280 T 0132

EXIT; %
END OF IFSTAT; %

00002280
00002280
02281 T 0142
02282 T 0143

00002266
28 IS 146 LONG, NEXT SEG 2

%
%
PROCEDURE CASESTAT; %
BEGIN%
DEFINE CASEHASH(N)=(N).[38:39] MOD MAXCASES; %

02283 T 0375
02284 T 0375
02285 T 0375
02286 T 0375
02287 T 0375

START OF SEGMENT ***** 29

INTEGER ARRAY CASETAB[0, MAXCASES]; %
INTEGER CASENUM, CASETYPE, NCASELABS, TEMPVARNUM, CONVAL, CONTYPE, C, T; %
BOOLEAN ZERO LAB, FIRST; %

02288 T 0000
02289 T 0001
02290 T 0001
02291 T 0001
02292 T 0001

CASENUM:=NUMCASES:=NUMCASES+1; MARGIN("CB", CASENUM); %

00002292
00002292

TEMPVARNUM:=NUMTEMPS:=NUMTEMPS+1; %
IF TEMPVARNUM>MAXTEMPS THEN ERROR(16); %
GEN(" BEGIN", 6, 3); GENID(" T", TEMPVARNUM, 2); GEN(" :=", 2, 6); %

02293 T 0021
02294 T 0023
02295 T 0025


```

NCASELABS:=NCASELABS+1;%
IF NCASELABS<MAXCASES THEN%
BEGIN%
  IF CONVAL=0 THEN%
    IF ZERO LAB THEN ERROR(31) ELSE ZERO LAB:=TRUE ELSE%
    BEGIN%
      T:=CASEHASH(CONVAL);%
      FOR C:=CASETAB[T] WHILE C#CONVAL AND C#0 DO%
        T:=IF T=0 THEN MAXCASES ELSE T-1;%
        IF C#0 THEN ERROR(31) ELSE CASETAB[T]:=CONVAL;%
      END;%
    END ELSE IF NCASELABS=MAXCASES THEN ERROR(30);%
    IF CURSY=COMMA THEN GEN(" OR",4,5);%
  END;%
END UNTIL CURSY#COMMA;%
GEN(" THEN",6,3);%
IF CURSY#COLON THEN BEGIN ERROR(26); SKIP(COLON) END;%
IF CURSY#COLON THEN INSYMBOL;%
STATEMENT;%
IF CURSY#SEMICOLON AND CURSY#ENDSY THEN%
BEGIN ERROR(21); SKIP(SEMICOLON) END;%
END;%
IF CURSY#SEMICOLON THEN GEN(" ELSE",6,3);%
END UNTIL CURSY#SEMICOLON;%
IF CURSY#ENDSY THEN BEGIN ERROR(24); SKIP(ENDSY) END;%
GEN(" END",5,4); MARGIN("CE",CASENUM);%

NUMTEMPS:=NUMTEMPS-1;%
INSYMBOL;%
END OF CASESTAT;%

```

```

%
%
PROCEDURE WHILESTAT;%

```

```

00002319
00002319
00002319
00002319

```

```

02320 T 0313
02321 T 0314
02322 T 0315
02323 T 0315
02324 T 0316
02325 T 0319
02326 T 0320
02327 T 0322
02328 T 0326
02329 T 0329
02330 T 0333

```

```

00002325
02331 T 0333

```

```

00002322
02332 T 0336

```

```

00002332
02333 T 0346

```

```

00002316
02334 T 0346

```

```

00002312
02335 T 0347

```

```

00002335
02336 T 0356

```

```

00002336
02337 T 0359

```

```

02338 T 0361
02339 T 0361

```

```

02340 T 0363
00002340

```

```

02341 T 0365
00002310

```

```

02342 T 0365
00002342

```

```

02343 T 0375
00002306

```

```

02344 T 0376
00002344

```

```

02345 T 0379
00002345
00002345
00002345

```

```

02346 T 0406
02347 T 0408
02348 T 0408

```

```

00002280

```

```

29 IS 423 LONG, NEXT SEG 2

```

```

02349 T 0375
02350 T 0375
02351 T 0375

```

```
BEGIN%
  LABEL STATM,EXIT;%
  GEN("WHILE",6,3);%
  INSYMBOL; BOOLEXP;%
```

```
IF CURSY#DOSY THEN%
BEGIN IF CURTYPE>0 THEN ERROR(19);%
  SKIP(DOSY);%
  IF CURSY#DOSY THEN%
  BEGIN IF CURTYPE=0 THEN ERROR(19);%
    GO TO IF SYMKIND[CURSY]=INITIAL THEN STATM ELSE EXIT;%
  END; END;%
```

```
GEN(" DO",4,5);%
```

```
INSYMBOL;%
STATM: STATEMENT;%
EXIT;%
END OF WHILESTAT;%
```

```
%
%
PROCEDURE REPEATSTAT;%
BEGIN%
  INTEGER REPNUM;%
  LABEL NEWTRY;%
  %
  REPNUM:=NUMREPS;=NUMREPS+1;%
  MARGIN(" R",REPNUM);%
  %
  GEN("DO",3,6); GEN("BEGIN",6,3);%
```

```
02352 T 0375
02353 T 0375
START OF SEGMENT ***** 30
02354 T 0000 00002354
02355 T 0009 00002355
00002355
00002355
00002355
00002355
00002355
00002355
00002355
00002355
00002355
00002355
00002355
00002355
00002355
00002355
00002355
00002355
```

```
02356 T 0112
02357 T 0113
02358 T 0115
02359 T 0116
02360 T 0117
02361 T 0119
02362 T 0125 00002360
00002357
02363 T 0125 00002363
02364 T 0134
02365 T 0134
02366 T 0135
02367 T 0136 00002352
30 IS 142 LONG, NEXT SEG 2
```

```
02368 T 0375
02369 T 0375
02370 T 0375
02371 T 0375
02372 T 0375
START OF SEGMENT ***** 31
02373 T 0000
02374 T 0000
02375 T 0000
02376 T 0001 00002376
00002376
02377 T 0020 00002377
00002377
```

```

DO BEGIN%
  INSYMBOL;%
NEWTRY: STATEMENT;%
  GEN(")",1,7);%

  IF CURSY=ELSESY THEN BEGIN ERROR(20); INSYMBOL; GO NEWTRY END;%
  IF SYMKIND[CURSY]=INITIAL THEN BEGIN ERROR(21); GO NEWTRY END;%

END UNTIL CURSY#SEMICOLON;%

IF CURSY#UNTILSY THEN%
BEGIN%
  ERROR(22);%
  WHILE CURSY#UNTILSY AND SYMKIND[CURSY]#INITIAL DO%
  BEGIN INSYMBOL; SKIP(UNTILSY) END;%

  IF CURSY#UNTILSY THEN GO TO NEWTRY;%
END;%

GEN(" END",5,4); GEN("UNTIL",6,3); MARGIN(" U",REPNUM);%

INSYMBOL; BOOLEXP;%

```

```

END OF REPEATSTAT;%

```

```

%
%
PROCEDURE FORSTAT;%
BEGIN%
  INTEGER VARTYPE,VARNUM,LLIM,ULIM;%

  BOOLEAN DOWN;%
  LABEL STATM;%
%

```

```

02378 T 0038
02379 T 0038
02380 T 0038
02381 T 0039 00002381
02382 T 0048 00002382
02383 T 0055 00002383
02384 T 0057 00002378
02385 T 0059
02386 T 0059
02387 T 0060
02388 T 0061
02389 T 0063 00002389
02390 T 0065
02391 T 0066 00002386
02392 T 0066 00002392
00002392
00002392
00002392
02393 T 0102 00002393
00002393
00002393
00002393
00002393
00002393
00002393
00002393
00002393
00002393
00002393
00002393
00002393
00002393
00002393
00002393
02394 T 0208 00002371
31 IS 211 LONG, NEXT SEG 2

02395 T 0375
02396 T 0375
02397 T 0375
02398 T 0375
02399 T 0375
START OF SEGMENT ***** 32
02400 T 0000
02401 T 0000
02402 T 0000

```



```

GENID("T",NUMTEMPS,2); GEN(" DO",4,5);%

STATM; STATEMENT;%
GEN(" END",5,4);%

NUMTEMPS:=NUMTEMPS-1;%
END OF FORSTAT;%

```

```

00002458
02459 T 0665
00002459
00002459
02460 T 0686
02461 T 0686
00002461
02462 T 0695
02463 T 0696
00002398
32 IS 703 LONG, NEXT SEG 2

```

```

%
%
PROCEDURE GOTOSTAT;%
BEGIN%
  INTEGER I;%

  INSYMBOL;%
  IF CURSY=INTCONST THEN%
  BEGIN I:=NUMLABS;%
    WHILE I>1 AND LABTAB[I].LABVAL<CURVAL DO I:=I-1;%
    IF I=0 THEN ERROR(15);%
    GEN("GO",3,6); GENID("L",CURVAL,4);%

    INSYMBOL;%
  END ELSE ERROR(10);%

END OF GOTOSTAT;%

```

```

02464 T 0375
02465 T 0375
02466 T 0375
02467 T 0375
02468 T 0375
START OF SEGMENT ***** 33
02469 T 0000
02470 T 0000
02471 T 0001
02472 T 0002
02473 T 0007
02474 T 0009
00002474
00002474
02475 T 0030
02476 T 0031
00002471
02477 T 0033
00002467
33 IS 36 LONG, NEXT SEG 2

```

```

%
%
PROCEDURE WITHSTAT;%
BEGIN%
  INTEGER STARTLEVEL,VERYFIRSTWITHSYM,I;%

  REAL D;%
  STARTLEVEL:=TOPLEVEL; VERYFIRSTWITHSYM:=NWITHSYMS;%
  DO BEGIN%
    INSYMBOL;%
    IF CURSY=IDENTIFIER THEN%
    BEGIN%
      SEARCH;%

      IF FOUND THEN%

```

```

02478 T 0375
02479 T 0375
02480 T 0375
02481 T 0375
02482 T 0375
START OF SEGMENT ***** 34
02483 T 0000
02484 T 0000
02485 T 0001
02486 T 0002
02487 T 0002
02488 T 0003
02489 T 0003
00002489
00002489
00002489
00002489
02490 T 0032

```

```

BEGIN%
  IF THISID.IDCLASS=VAR THEN%
  BEGIN%
    VARIABLE;%
    IF CURTYPE>0 THEN%
    IF TYPETAB1[CURTYPE].FORM#RECORD THEN ERROR(98);%
    IF SIMPLEVARIABLE THEN%
    BEGIN PUTSYM("("); INSIDEBRACKETS:=TRUE END;%

    IF TOPLEVEL<MAXLEVEL THEN%
    BEGIN%
      TOPLEVEL:=TOPLEVEL+1;%
      D.NAMETAB:=TYPETAB1[CURTYPE].RECTAB;%
      D.RECTYPE:=CURTYPE;%
      D.NUMPTRSINWITH:=NUMPOINTERS;%
      D.FIRSTWITHSYM:=NWITHSYMS;%
      D.BRACKETSINWITH:=REAL(INSIDEBRACKETS);%
      IF NWITHSYMS+NUMSYMS>MAXWITHSYMS THEN ERROR(63) ELSE%
      FOR I:=1 STEP 1 UNTIL NUMSYMS DO%
      BEGIN%
        WITHTAB[NWITHSYMS]:=SYMTAB[I];%
        NWITHSYMS:=NWITHSYMS+1;%
      END;%

      D.LASTWITHSYM:=NWITHSYMS-1;%
      DISPLAY[TOPLEVEL]:=D;%
      END ELSE ERROR(84);%

      END ELSE BEGIN ERROR(8); INSYMBOL END;%

      END ELSE BEGIN ERROR(1); INSYMBOL END;%

      END ELSE BEGIN ERROR(9); INSYMBOL END;%

      NUMSYMS:=0;%
      NUMPOINTERS := 0;%
      END UNTIL CURSY#COMMA;%

      IF CURSY#DOSY THEN%
      BEGIN ERROR(19); SKIP(DOSY);%
      IF CURSY=DOSY THEN INSYMBOL;%
      END ELSE INSYMBOL;%

      STATEMENT;%
      TOPLEVEL:=STARTLEVEL; NWITHSYMS:=VERYFIRSTWITHSYM;%
      END OF WITHSTAT;%

```

```

02491 T 0032
02492 T 0032
02493 T 0034
02494 T 0034
02495 T 0035
02496 T 0035
02497 T 0039
02498 T 0039
00002498
00002498
00002498
00002498
02499 T 0048
02500 T 0048
02501 T 0049
02502 T 0050
02503 T 0053
02504 T 0054
02505 T 0056
02506 T 0058
02507 T 0060
02508 T 0062
02509 T 0064
02510 T 0064
02511 T 0065
02512 T 0066
00002509
02513 T 0069
02514 T 0071
02515 T 0072
00002500
02516 T 0073
00002493
00002516
02517 T 0075
00002491
00002517
02518 T 0077
00002488
00002518
02519 T 0079
02520 T 0079
02521 T 0080
00002485
02522 T 0081
02523 T 0082
02524 T 0084
02525 T 0086
00002523
02526 T 0087
02527 T 0087
02528 T 0089
00002481

```

```

%
%
PROCEDURE STATEMENT;%
BEGIN%
  INTEGER I;%

  LABEL LABFOUND;%
%
  IF CURSY=INTCONST THEN          % *** LABELED STATEMENT ***
  BEGIN%
    FOR I:=FIRSTLAB STEP 1 UNTIL NUMLABS DO%
      IF LABTAB[I],LABVAL=CURVAL THEN%
        BEGIN IF LABTAB[I],LABDEF=1 THEN ERROR(31);%
          LABTAB[I],LABDEF:=1;%
          GO TO LABFOUND;%
        END;%

        ERROR(15);%
LABFOUND: GENID("L",CURVAL,4); GEN("!",1,7);%

        INSYMBOL;%
        IF CURSY#COLON THEN%
          BEGIN ERROR(26);%
            SKIP(COLON); IF CURSY=COLON THEN INSYMBOL;%
          END ELSE INSYMBOL;%
        END;%
%
  COMMENT *** START OF STATEMENT *** ;%
%
  IF CURSY=IDENTIFIER THEN%
  BEGIN%
    SEARCH;%

    IF FOUND THEN%
    BEGIN%
      IF THISID.IDCLASS=VAR OR%
        THISID.IDCLASS=CONST AND BOOLEAN(THISID.FORMAL) OR%
        THISID.IDCLASS=FUNC THEN ASSIGNMENT ELSE%
      IF THISID.IDCLASS=PROC THEN%
      BEGIN%
        IF THISLEVEL=0 THEN          % *** INTRINSIC PROCEDURE ***
        BEGIN%
          IF CURNAME1="5WRITE" THEN PWRITE(FALSE) ELSE%
          IF CURNAME1="7WRITE" AND%
            CURNAME2="00000N" THEN PWRITE(TRUE) ELSE%
          IF CURNAME1="400READ" THEN PREAD(FALSE) ELSE%
          IF CURNAME1="6READLN" THEN PREAD(TRUE) ELSE%
          IF CURNAME1="400PAGE" THEN FILEHANDLING(5) ELSE%
          IF CURNAME1="300GET" THEN FILEHANDLING(2) ELSE%
          IF CURNAME1="300PUT" THEN FILEHANDLING(1) ELSE%
          IF CURNAME1="50RESET" THEN FILEHANDLING(3) ELSE%

```

```

02529 T 0375
02530 T 0375
02531 T 0375
02532 T 0375
02533 T 0375
START OF SEGMENT ***** 35
02534 T 0000
02535 T 0000
02536 T 0000
02537 T 0000
02538 T 0001
02539 T 0002
02540 T 0003
02541 T 0006
02542 T 0009
02543 T 0009
00002540
02544 T 0012
02545 T 0012
00002545
00002545
02546 T 0034
02547 T 0034
02548 T 0035
02549 T 0036
02550 T 0039
00002548
02551 T 0040
00002537
02552 T 0040
02553 T 0040
02554 T 0040
02555 T 0040
02556 T 0040
02557 T 0041
00002557
00002557
00002557
00002557
02558 T 0070
02559 T 0070
02560 T 0070
02561 T 0072
02562 T 0074
02563 T 0077
02564 T 0078
02565 T 0079
02566 T 0080
02567 T 0080
02568 T 0082
02569 T 0084
02570 T 0087
02571 T 0091
02572 T 0095
02573 T 0099
02574 T 0103
02575 T 0109

```

```

IF CURNAME1="7REWRIT" AND%
  CURNAME2="000000E" THEN FILEHANDLING(4) ELSE%
IF CURNAME1="3000NEW" THEN NEWDISP ELSE%
IF CURNAME1="7DISPOS" AND%
  CURNAME2="000000E" THEN NEWDISP ELSE%
IF CURNAME1="400PACK" THEN PACK ELSE%
IF CURNAME1="6UNPACK" THEN UNPACK ELSE ERROR(0);%
END ELSE PASSPARAMS;%

WRITEEXPR;%

END ELSE BEGIN ERROR(13); SKIP(99) END;%

END ELSE BEGIN ERROR(1); ASSIGNMENT END;%

END OF IDENTIFIER ELSE%

IF CURSY=BEGINSY THEN COMPSTAT ELSE%
IF CURSY=IFSY THEN IFSTAT ELSE%
IF CURSY=CASESY THEN CASESTAT ELSE%
IF CURSY=WHILESY THEN WHILESTAT ELSE%
IF CURSY=REPEATSY THEN REPEATSTAT ELSE%
IF CURSY=FORSY THEN FORSTAT ELSE%
IF CURSY=WITHSY THEN WITHSTAT ELSE%
IF CURSY=GOTOSY THEN GOTOSTAT ELSE%
IF SYMKIND[CURSY]≠TERMINAL THEN%
BEGIN ERROR(13); INSYMBOL; SKIP(SEMICOLON) END;%

END OF STATEMENT;%

```

```

02576 T 0111
02577 T 0113
02578 T 0116
02579 T 0119
02580 T 0122
02581 T 0124
02582 T 0128
02583 T 0135
00002566
02584 T 0136
00002584
00002584
00002584
00002584
00002584
00002584
00002584
02585 T 0215
00002564
00002585
02586 T 0217
00002559
00002586
02587 T 0219
00002556
02588 T 0219
02589 T 0221
02590 T 0223
02591 T 0226
02592 T 0228
02593 T 0230
02594 T 0232
02595 T 0235
02596 T 0237
02597 T 0238
00002597
02598 T 0241
00002532

```

35 IS 244 LONG, NEXT SEG 2

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X
X
X
X
X
X
X
X
X
X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X
X
REAL VALX1,VALX2;%
INTEGER TYPEX1,TYPEX2;%
BOOLEAN PACKED;%
X
PROCEDURE FIELDLIST(RECTAB,FIRSTADDR, LASTADDR);%
VALUE RECTAB,FIRSTADDR;%
INTEGER RECTAB,FIRSTADDR, LASTADDR;%
FORWARD;%
X
DEFINE SUBRANGE#                                     %*** SUBRANGE DECLARATION ***
BEGIN                                               %*****
    CONSTANT(VALX1,TYPEX1);%
    IF TYPETAB1[TYPEX1].FORM>CHAR THEN ERROR(11);%
    IF CURSY#DOUBLEDOT THEN ERROR(53);%
    INSYMBOL;%
    CONSTANT(VALX2,TYPEX2);%
    IF TYPEX1>0 AND TYPEX2>0 THEN%
        IF TYPEX1#TYPEX2 THEN ERROR(11) ELSE%
            IF VALX1>VALX2 THEN ERROR(54);%
    T1:=TYPETAB1[TYPEX1].FORM; IF T1=SYMBOLIC THEN T1:=SUBTYPE;%
    NEWTYPE; TTYPE:=TYPEINDEX;%
    T1.SIZE:=TSIZE:=1; T1.STRUCT:=0; T1.MAINTYPE:=TYPEX1;%
    TYPETAB1[TYPEINDEX]:=T1;%
    TYPETAB2[TYPEINDEX]:=VALX1; TYPETAB3[TYPEINDEX]:=VALX2;%
END OF SUBRANGE#;%
X
X
PROCEDURE TYPEDECL(TTYPE,TSIZE);%
INTEGER TTYPE,TSIZE;%
BEGIN%
    PROCEDURE TYPERR(ERRNUM,TTYPE,TSIZE);%
        VALUE ERRNUM;%
        INTEGER ERRNUM,TTYPE,TSIZE;%
        BEGIN ERROR(ERRNUM);%
            TTYPE:=TSIZE:=0;%
        END;%
X
INTEGER RECINX,ARRSTRUCT, TX, SX, T1, T2, T3, T, N;%
BOOLEAN FIRST;%

```

```

02600 T 0375
02601 T 0375
02602 T 0375
02603 T 0375
02604 T 0375
02605 T 0375
02606 T 0375
02607 T 0375
02608 T 0375
02609 T 0375
02610 T 0375
02611 T 0375
02612 T 0375
02613 T 0375
02614 T 0375
02615 T 0375
02616 T 0375
02617 T 0375
02618 T 0375
02619 T 0375
02620 T 0375
02621 T 0375
02622 T 0375
02623 T 0375
02624 T 0375
02625 T 0375
02626 T 0375
02627 T 0375
02628 T 0375
02629 T 0375
02630 T 0375
02631 T 0375
02632 T 0375
02633 T 0375
02634 T 0375
02635 T 0375
02636 T 0375
02637 T 0375
02638 T 0375
02639 T 0375
02640 T 0375
02641 T 0375
02642 T 0375
START OF SEGMENT ***** 36
02643 T 0000
02644 T 0000
02645 T 0000
02646 T 0000
02647 T 0002
00002645
02648 T 0002
02649 T 0002
02650 T 0002

```



```

IF CURSY=IDENTIFIER THEN%
BEGIN%
  NEWNAME(CURNAME1,CURNAME2,CURLEVEL);%

  T3.INFO:=N; NAMETAB3(CURLEVEL,THISINDEX):=T3;%
  N:=N+1; INSYMBOL;%
  END ELSE ERROR(9);%

END UNTIL CURSY#COMMA;%

IF CURSY#RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;%

T1:=SYMBOLIC; T1.STRUCT:=0;%
T1.SIZE:=TSIZE:=1; TTYPE:=TYPEINDEX;%
TYPETAB1[TYPEINDEX]:=T1;%
TYPETAB2[TYPEINDEX]:=0; TYPETAB3[TYPEINDEX]:=N-1;%
IF CURSY#RPAR THEN INSYMBOL;%
END ELSE%

%
IF CURSY=ARROW THEN
BEGIN
  INSYMBOL;%
  IF CURSY=IDENTIFIER THEN%
  BEGIN%
    NEWTYPE; TTYPE:=TYPEINDEX; T1:=POINTERS;%

    T1.SIZE:=TSIZE:=1; T1.STRUCT:=0;%
    TYPETAB1[TYPEINDEX]:=T1;%
    SEARCH;%

    IF FOUND THEN%
    BEGIN%
      IF THISID.IDCLASS=TYPES THEN%
        TYPETAB1[TYPEINDEX].POINTTYPE:=THISID.TYPE ELSE%
        TYPERR(7,TTYPE,TSIZE);%
      END ELSE%

      BEGIN%
        IF NUMPNTRS<MAXPNTRS THEN NUMPNTRS:=NUMPNTRS+1 ELSE ERROR(52);
        PNTRTAB1[NUMPNTRS]:=CURNAME1; PNTRTAB2[NUMPNTRS]:=CURNAME2;%
        PNTRTAB3[NUMPNTRS]:=TYPEINDEX;%
      END;%

      INSYMBOL;%
    END ELSE TYPERR(9,TTYPE,TSIZE);%

  END OF POINTER DECLARATION ELSE%

  BEGIN%
    IF CURSY=PACKEDSY THEN BEGIN PACKED:=TRUE; INSYMBOL END;%

```

```

02673 T 0477
02674 T 0478
02675 T 0478
00002675
00002675
02676 T 0501
02677 T 0505
02678 T 0507
00002678
02679 T 0508
00002671
02680 T 0509
00002680
02681 T 0512
02682 T 0514
02683 T 0518
02684 T 0519
02685 T 0522
02686 T 0524
00002668
02687 T 0524
02688 T 0524
02689 T 0525
02690 T 0526
02691 T 0526
02692 T 0527
02693 T 0527
00002693
00002693
02694 T 0534
02695 T 0538
02696 T 0540
00002696
00002696
00002696
00002696
02697 T 0569
02698 T 0569
02699 T 0569
02700 T 0571
02701 T 0574
02702 T 0577
00002698
02703 T 0577
02704 T 0577
02705 T 0581
02706 T 0584
02707 T 0585
00002703
02708 T 0585
02709 T 0585
00002692
02710 T 0588
00002689
02711 T 0588
02712 T 0589
00002712

```

```

%
IF CURSY=ARRAYSY THEN          %*** ARRAY DECLARATION ***
BEGIN                          %*****
INSYMBOL;%
IF CURSY#LBRACKET THEN ERROR(47) ELSE INSYMBOL;%
T:=0; FIRST:=TRUE;%
DO BEGIN%
IF FIRST THEN FIRST:=FALSE ELSE INSYMBOL;%
TYPEDECL(TX,SX);%
IF TX>0 THEN%
BEGIN%
IF TYPETAB1[TX].FORM>CHAR THEN ERROR(48);%
T1:=ARRAYS; T1.INXTYPE:=TX; T1.ARRTYPE:=T;%
T2:=TYPETAB2[TX]; T3:=TYPETAB3[TX];%
IF T3-T2>1022 THEN ERROR(61);%
T1.SIZE:=MIN(1023,T3-T2+1);%
NEWTYP;%
TYPETAB1[TYPEINDEX]:=T1;%
TYPETAB2[TYPEINDEX]:=T2; TYPETAB3[TYPEINDEX]:=T3;%
T:=TYPEINDEX;%
END;%
END UNTIL CURSY#COMMA;%
IF CURSY#RBRACKET THEN ERROR(59) ELSE INSYMBOL;%
IF CURSY#OFSY THEN BEGIN ERROR(18); SKIP(OFSY) END;%
INSYMBOL;%
TYPEDECL(TX,SX);%
IF TYPETAB1[TX].FORM#FILES THEN ERROR(60);%
ARRSTRUCT:=TYPETAB1[TX].STRUCT;%
WHILE T>0 DO%
BEGIN%
T1:=TYPETAB1[T]; T3:=T1.ARRTYPE;%
T1.ARRTYPE:=TX; T1.STRUCT:=ARRSTRUCT+1;%
T1.SIZE:=SX:=MIN(1024,SX*T1.SIZE);%
TYPETAB1[T]:=T1; TX:=T; T:=T3;%
END;%
TTYPE:=TX; TSIZE:=SX;%
END OF ARRAY DECLARATION ELSE%

```

```

%
IF CURSY=FILESY THEN          %*** FILE DECLARATION ***
BEGIN                          %*****
INSYMBOL;%
IF CURSY#OFSY THEN%
BEGIN ERROR(18);%
IF CURSY#IDENTIFIER THEN INSYMBOL;%
END ELSE INSYMBOL;%
TYPEDECL(TX,SX);%
IF TX>0 THEN%
BEGIN T:=TYPETAB1[TX];%
IF T.FORM#FILES THEN ERROR(50) ELSE%

```

```

02713 T 0591
02714 T 0591
02715 T 0592
02716 T 0592
02717 T 0593
02718 T 0596
02719 T 0597
02720 T 0598
02721 T 0600
02722 T 0601
02723 T 0602
02724 T 0602
02725 T 0605
02726 T 0609
02727 T 0611
02728 T 0614
02729 T 0619
00002729
00002729
02730 T 0624
02731 T 0625
02732 T 0628
02733 T 0628
00002728
02734 T 0628
00002719
02735 T 0630
02736 T 0633
00002736
02737 T 0635
02738 T 0636
02739 T 0637
02740 T 0640
02741 T 0641
02742 T 0643
02743 T 0643
02744 T 0645
02745 T 0650
02746 T 0655
02747 T 0658
00002742
02748 T 0660
02749 T 0662
00002715
02750 T 0662
02751 T 0662
02752 T 0663
02753 T 0663
02754 T 0664
02755 T 0665
02756 T 0666
02757 T 0668
00002755
02758 T 0669
02759 T 0670
02760 T 0670
02761 T 0672

```

```

        IF T.STRUCT>1 THEN ERROR(49));%
    END;%

    NEWTYPE; TTYPE:=TYPEINDEX;%

    T1:=IF T.FORM=CHAR THEN TEXTFILE ELSE FILES;%
    T1.SIZE:=TSIZE:=SX; T1.FILETYPE:=TX;%
    T1.STRUCT:=1;%
    TYPETAB1[TYPEINDEX]:=T1;%
    END OF FILE DECLARATION ELSE%

```

```

02762 T 0674
02763 T 0677
02764 T 0677
02765 T 0683
02766 T 0687
02767 T 0691
02768 T 0693
02769 T 0694

```

```

00002760
00002764
00002764

```

*

```

    IF CURSY=SETSY THEN
    BEGIN
        INSYMBOL;%
        IF CURSY<=0 THEN%
        BEGIN ERROR(18));%
            IF CURSY>CHARCONST THEN INSYMBOL;%
        END ELSE INSYMBOL;%

        TYPEDECL(TX,SX);%
        IF TX>0 THEN%
        BEGIN%
            IF TYPETAB1[TX].FORM>CHAR THEN ERROR(48) ELSE%
            IF TYPETAB2[TX]<0 OR TYPETAB3[TX]>38 THEN ERROR(51));%
        END;%
    
```

```

02770 T 0694
02771 T 0694
02772 T 0695
02773 T 0696
02774 T 0696
02775 T 0697
02776 T 0698
02777 T 0700

```

```
00002752
```

```
00002775
```

```
NEWTYPE; TTYPE:=TYPEINDEX;%
```

```

    T1:=SET; T1.SETTYPE:=TX; T1.STRUCT:=0;%
    T1.SIZE:=TSIZE:=1; TYPETAB1[TYPEINDEX]:=T1;%
    TYPETAB2[TYPEINDEX]:=TYPETAB2[TX];%
    TYPETAB3[TYPEINDEX]:=TYPETAB3[TX];%
    END OF SET DECLARATION ELSE%

```

```

02778 T 0701
02779 T 0702
02780 T 0703
02781 T 0703
02782 T 0706
02783 T 0710

```

```
00002780
```

```
02784 T 0710
```

```
00002784
00002784
```

```

02785 T 0716
02786 T 0720
02787 T 0724
02788 T 0725
02789 T 0727

```

```
00002772
```

*

```

    IF CURSY=RECORDSY THEN
    BEGIN
        IF LASTREC=1>CURLEVEL THEN LASTREC:=LASTREC-1 ELSE ERROR(55));%
        RECINX:=LASTREC;%
        BLOCKTAB[RECINX]:=NUMBLOCKS:=NUMBLOCKS+1;%
        INSYMBOL;%
        FIELDLIST(RECINX,0,SX);%
        IF SX>1022 THEN BEGIN ERROR(56); SX:=1022 END;%
    
```

```

02790 T 0727
02791 T 0727
02792 T 0728
02793 T 0729
02794 T 0733
02795 T 0734
02796 T 0736
02797 T 0736
02798 T 0738

```

```
00002798
```

```
NEWTYPE; TTYPE:=TYPEINDEX;%
```

```
02799 T 0740
```

```
00002799
00002799
```

```

    T1:=RECORD; T1.RECTAB:=RECINX; T1.STRUCT:=1;%
    T1.SIZE:=TSIZE:=SX; TYPETAB1[TYPEINDEX]:=T1;%
    TYPETAB2[TYPEINDEX]:=0; TYPETAB3[TYPEINDEX]:=SX-1;%
    IF CURSY<=ENDSY THEN BEGIN ERROR(24); SKIP(ENDSY) END;%

```

```

02800 T 0746
02801 T 0751
02802 T 0754
02803 T 0757

```

```
00002803
```

```

    IF CURSY=ENDSY THEN INSYMBOL;%
    END ELSE BEGIN ERROR(4); SKIP(99) END;%

```

```

02804 T 0760
02805 T 0762

```

END;%
END OF TYPEDECL;%

00002792
00002805
02806 T 0764
00002711
02807 T 0764
00002641
36 IS 777 LONG, NEXT SEG 2

```
%  
%  
PROCEDURE FIELDLIST(RECTAB,FIRSTADDR, LASTADDR);%  
VALUE RECTAB,FIRSTADDR;%  
INTEGER RECTAB,FIRSTADDR, LASTADDR;%  
BEGIN%  
  INTEGER ARRAY ILIST(0:LISTLENGTH);%  
  
  INTEGER LISTINX;%  
  INTEGER CASETYPE, ADDR, MAXADDR, INDEX, CTYPE, TX, SX, T1, T3, LLIM, ULIM, I;%  
  BOOLEAN FIRST;%  
  REAL CVAL;%  
  LABEL CASETYPEID, CASEPART, EXIT;%  
%  
  ADDR:=FIRSTADDR;%  
  DO BEGIN%  
    WHILE CURSY=SEMICOLON DO INSYMBOL;%  
    IF CURSY=CASESY THEN GO TO CASEPART;%  
    IF CURSY=IDENTIFIER THEN%  
      BEGIN%  
        LISTINX:=0; FIRST:=TRUE;%  
        DO BEGIN%  
          IF FIRST THEN FIRST:=FALSE ELSE INSYMBOL;%  
          IF CURSY=IDENTIFIER THEN%  
            BEGIN%  
              IF LISTINX>=LISTLENGTH THEN BEGIN ERROR(37); LISTINX:=0 END;%  
  
              LISTINX:=LISTINX+1;%  
              NEWNAME(CURNAME1,CURNAME2,RECTAB);%  
  
              ILIST[LISTINX]:=THISINDEX;%  
              INSYMBOL;%  
            END ELSE%  
  
            BEGIN ERROR(9);%  
              IF CURSY#COMMA THEN INSYMBOL;%  
            END;%  
  
          END UNTIL CURSY#COMMA;%  
  
          IF CURSY#COLON THEN BEGIN ERROR(26); SKIP(COLON) END;%  
  
          INSYMBOL;%  
          TYPEDECL(TX,SX);%  
          IF TX>0 THEN IF TYPETAB1[TX].FORM>FILES THEN ERROR(57);%  
        END%  
      END%  
    END%  
  END%  
END%
```

```
02808 T 0375  
02809 T 0375  
02810 T 0375  
02811 T 0375  
02812 T 0375  
02813 T 0375  
02814 T 0375  
START OF SEGMENT ***** 37  
02815 T 0001  
02816 T 0001  
02817 T 0001  
02818 T 0001  
02819 T 0001  
02820 T 0001  
02821 T 0001  
02822 T 0002  
02823 T 0003  
02824 T 0005  
02825 T 0006  
02826 T 0007  
02827 T 0007  
02828 T 0009  
02829 T 0010  
02830 T 0012  
02831 T 0013  
02832 T 0013  
  
00002832  
02833 T 0016  
02834 T 0017  
  
00002834  
00002834  
02835 T 0040  
02836 T 0041  
02837 T 0042  
  
00002831  
02838 T 0042  
02839 T 0043  
02840 T 0045  
  
00002838  
02841 T 0045  
  
00002828  
02842 T 0046  
  
00002842  
02843 T 0049  
02844 T 0049  
02845 T 0050
```

```

T3.IDCLASS:=VAR; T3.TYPE:=TX;%
FOR I:=1 STEP 1 UNTIL LISTINX DO%
BEGIN%
  T3.INFO:=ADDR; ADDR:=MIN(ADDR+SX,1024);%
  NAMETAB3[RECTAB,ILIST[I]]:=T3;%
END;%

END;%

END UNTIL CURSY#SEMICOLON;%

LASTADDR:=ADDR;%
GO TO EXIT;%

%
CASEPART;%
LISTINX:=0; LASTADDR:=ADDR; INDEX:=-1;%
INSYMBOL;%
IF CURSY=IDENTIFIER THEN%
BEGIN%
  SEARCH;%

IF FOUND AND THISID.IDCLASS=TYPES THEN GO TO CASETYPEID;%
NEWNAME(CURNAME1,CURNAME2,RECTAB); INDEX:=THISINDEX;%

INSYMBOL;%
IF CURSY#COLON THEN ERROR(26);%
INSYMBOL;%
IF CURSY=IDENTIFIER THEN%
BEGIN%
  SEARCH;%

IF FOUND THEN%
BEGIN%
  IF THISID.IDCLASS=TYPES THEN%
  BEGIN%
CASETYPEID: CASETYPE:=THISID.TYPE; T1:=TYPETAB1[CASETYPE];%
LLIM:=TYPETAB2[CASETYPE]; ULIM:=TYPETAB3[CASETYPE];%
IF T1.FORM>CHAR THEN ERROR(48);%
IF INDEX>0 THEN%
BEGIN%
  T3.IDCLASS:=VAR; T3.TYPE:=CASETYPE; T3.INFO:=ADDR;%
  ADDR:=LASTADDR:=ADDR+1; NAMETAB3[RECTAB,INDEX]:=T3;%
END;%

INSYMBOL;%
END ELSE BEGIN ERROR(7); SKIP(QFSY) END;%

END ELSE BEGIN ERROR(1); SKIP(QFSY) END;%

```

```

02846 T 0054
02847 T 0058
02848 T 0059
02849 T 0059
02850 T 0064
02851 T 0066
00002848
02852 T 0068
00002826
02853 T 0068
00002822
02854 T 0070
02855 T 0071
02856 T 0073
02857 T 0073
02858 T 0073
02859 T 0075
02860 T 0076
02861 T 0077
02862 T 0077
00002862
00002862
00002862
00002862
02863 T 0106
02864 T 0108
00002864
00002864
02865 T 0131
02866 T 0131
02867 T 0133
02868 T 0134
02869 T 0135
02870 T 0135
00002870
00002870
00002870
00002870
02871 T 0164
02872 T 0164
02873 T 0164
02874 T 0166
02875 T 0166
02876 T 0169
02877 T 0171
02878 T 0173
02879 T 0174
02880 T 0175
02881 T 0180
02882 T 0184
00002879
02883 T 0184
02884 T 0184
00002874
00002884
02885 T 0186
00002872

```

```

END ELSE BEGIN ERROR(9); SKIP(OFSY) END;%

END ELSE BEGIN ERROR(9); SKIP(OFSY) END;%

IF CURSY#OFSY THEN BEGIN ERROR(18); SKIP(RPAR) END;%

IF CURSY=OFSY THEN INSYMBOL;%
IF CASETYPE=0 THEN BEGIN LLIM:=-MAXINT; ULIM:=MAXINT END;%

DO BEGIN%
  WHILE CURSY=SEMICOLON DO INSYMBOL;%
  IF CURSY<CHARCONST OR CURSY=PLUS OR CURSY=MINUS THEN%
  BEGIN%
    FIRST:=TRUE;%
    DO BEGIN%
      IF FIRST THEN FIRST:=FALSE ELSE INSYMBOL;%
      CONSTANT(CVAL,CTYPE);%

```

```

IF CTYPE>0 THEN%
BEGIN%
  IF CASETYPE=0 THEN CASETYPE:=CTYPE ELSE%
  IF CVAL<LLIM OR CVAL>ULIM THEN ERROR(14) ELSE%
  CHECKTYPES(CASETYPE,CTYPE);%

```

```

IF LISTINX>LISTLENGTH THEN BEGIN ERROR(30); LISTINX:=0 END;%

LISTINX:=LISTINX+1;%
ILIST[LISTINX]:=CVAL; I:=1;%
WHILE ILIST[I]#CVAL DO I:=I+1;%
IF I<LISTINX THEN ERROR(31);%

```

```

02886 T 0188 00002885
00002869
00002886
02887 T 0190 00002861
00002887
02888 T 0192 00002888
02889 T 0195
02890 T 0197 00002890
02891 T 0200
02892 T 0201
02893 T 0205
02894 T 0207
02895 T 0208
02896 T 0209
02897 T 0209
02898 T 0211 00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
00002898
02899 T 0299
02900 T 0299
02901 T 0300
02902 T 0302
02903 T 0305 00002903
00002903
00002903
00002903
00002903
00002903
00002903
00002903
00002903
02904 T 0365 00002904
02905 T 0367
02906 T 0369
02907 T 0371
02908 T 0374

```

```

END;%
END UNTIL CURSY#COMMA;%
IF CURSY#COLON THEN BEGIN ERROR(26); SKIP(LPAR) END;%
IF CURSY#COLON THEN INSYMBOL;%
IF CURSY#LPAR THEN%
BEGIN%
INSYMBOL; FIELDLIST(RECTAB,ADDR,MAXADDR);%
IF MAXADDR>LASTADDR THEN LASTADDR:=MAXADDR;%
IF CURSY#RPAR THEN BEGIN ERROR(46); SKIP(RPAR) END;%

INSYMBOL;%
END ELSE ERROR(58);%

END;%

END UNTIL CURSY NEQ SEMICOLON; %

EXIT;%
END OF FIELDLIST;%

```

```

02909 T 0376
00002900
02910 T 0376
00002896
02911 T 0377
00002911
02912 T 0380
02913 T 0382
02914 T 0382
02915 T 0383
02916 T 0385
02917 T 0387
00002917
02918 T 0390
02919 T 0390
00002914
02920 T 0391
00002894
02921 T 0391
00002891
02922 T 0393
02923 T 0393
00002813
37 IS 401 LONG, NEXT SEG 2

```



```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X
X
X
X
X
X
X
X
X
X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X
X
X
PROCEDURE DECLAREVARS(PARAM,TAB,FIRST,LAST,LEVEL);%
VALUE PARAM,FIRST,LAST,LEVEL;%
INTEGER ARRAY TAB[0];%
INTEGER FIRST,LAST,LEVEL;%
BOOLEAN PARAM;%
BEGIN%
  INTEGER LEVEL1000,TYP,NAM,NAMTAB,T1,I,J,REQSIZE;%

  BOOLEAN REALVAR,ARRAYVAR,FIRSTDIM,EXTFILE;%
  ALPHA FNAME;%
  INTEGER FNLENGTH,FNSTART;%
X
  LEVEL1000:=LEVEL*1000;%
  FOR I:=FIRST STEP 1 UNTIL LAST DO%
  BEGIN%
    NAM:=TAB[I],[9:10]; NAMTAB:=NAMETAB3[LEVEL,NAM];%
    TYP:=NAMTAB.TYPE; T1:=TYPETAB1[TYP];%
    IF NAMTAB.IDCLASS GEQ FUNC THEN%
    BEGIN%
      IF REALVAR OR ARRAYVAR THEN%
      BEGIN%
        GEN(")",1,7);%

        REALVAR:=ARRAYVAR:=FALSE;%
      END;%

      IF NAMTAB.IDCLASS=FUNC THEN GEN("REAL",5,4);%

      GEN("PROCEDU",8,1);%

      GENID("V",LEVEL1000+NAM,5); GEN(")",1,7);%

    END ELSE%

    IF T1,STRUCT=0 THEN
      X*** SIMPLE TYPE ***
    BEGIN%
      IF ARRAYVAR THEN BEGIN GEN(")",1,7); ARRAYVAR:=FALSE END;%

      IF REALVAR THEN GEN(")",1,7) ELSE%
      BEGIN GEN("REAL",5,4); REALVAR:=TRUE END;%

```

```

02925 T 0375
02926 T 0375
02927 T 0375
02928 T 0375
02929 T 0375
02930 T 0375
02931 T 0375
02932 T 0375
02933 T 0375
02934 T 0375
02935 T 0375
02936 T 0375
02937 T 0375
02938 T 0375
02939 T 0375
02940 T 0375
02941 T 0375
02942 T 0375
02943 T 0375
02944 T 0375
START OF SEGMENT ***** 38
02945 T 0000
02946 T 0000
02947 T 0000
02948 T 0000
02949 T 0000
02950 T 0001
02951 T 0002
02952 T 0002
02953 T 0003
02954 T 0007
02955 T 0008
02956 T 0009
02957 T 0010
02958 T 0010
00002958
02959 T 0019
02960 T 0020
00002957
02961 T 0020
00002961
02962 T 0031
00002962
02963 T 0040
00002963
00002963
02964 T 0062
00002955
02965 T 0062
02966 T 0066
02967 T 0066
00002967
00002967
02968 T 0077
00002968
02969 T 0087
00002969

```

```

GENID("V",LEVEL1000+NAM,5);%
END ELSE%
BEGIN%
  IF REALVAR THEN BEGIN GEN(";",1,7); REALVAR:=FALSE END;%

  IF T1.FORM<FILES THEN          %*** ARRAY/RECORD ***
  BEGIN%
    IF ARRAYVAR THEN GEN(";",1,7) ELSE%
      BEGIN GEN("ARRAY",6,3); ARRAYVAR:=TRUE END;%

    GENID("V",LEVEL1000+NAM,5); GEN("[",1,7);%

    FIRSTDIM:=TRUE;%
    DO BEGIN%
      IF FIRSTDIM THEN FIRSTDIM:=FALSE ELSE GEN(";",1,7);%
      GENINT(TYPETAB2[TYP]);%

      IF NOT PARAM THEN%
        BEGIN GEN(";",1,7); GENINT(TYPETAB3[TYP]) END;%

      TYP:=IF T1.FORM=ARRAYS THEN T1.ARRTYPE ELSE REALTYPE;%
      T1:=TYPETAB1[TYP];%
    END UNTIL T1.STRUCT=0;%

    GEN("]",1,7);%
  END ELSE%
  BEGIN          %*** FILE ***
    IF REALVAR OR ARRAYVAR THEN%
      BEGIN GEN(";",1,7); REALVAR:=ARRAYVAR:=FALSE END;%

    IF T1.FORM=TEXTFILE AND NOT PARAM THEN%
    BEGIN%
      IF NUMFILES>=MAXFILES THEN ERROR(97);%
      ELSE NUMFILES:=NUMFILES+1;%
      FILETAB[NUMFILES]:=NAM;%
    END;%

    EXTFILE:=FALSE;%
    FNAME:=NAMETAB1[LEVEL,NAM];%
    FNLENGTH := FNAME.NAMELENGTH; FNSTART := 8-FNLENGTH;%
    IF FNLENGTH LEQ 6 THEN %
    BEGIN%

```

```

02970 T 0097 00002969
02971 T 0109 00002970
02972 T 0109 00002966
02973 T 0112 00002973
02974 T 0122 00002973
02975 T 0123
02976 T 0124 00002976
02977 T 0134 00002977
02978 T 0144 00002977
02979 T 0165 00002978
02980 T 0166
02981 T 0167 00002981
02982 T 0179 00002982
02983 T 0223 00002982
02984 T 0223 00002984
02985 T 0277 00002984
02986 T 0281 00002984
02987 T 0282 00002984
02988 T 0283 00002980
02989 T 0292 00002988
02990 T 0292 00002975
02991 T 0293
02992 T 0294 00002992
02993 T 0304 00002992
02994 T 0306
02995 T 0307
02996 T 0309
02997 T 0311
02998 T 0312 00002994
02999 T 0312
03000 T 0313
03001 T 0314
03002 T 0317
03003 T 0318

```

```

FOR J:=1 STEP 1 UNTIL NUMEXTFILES DO%
  IF FNAME=EXTFILETAB[J] THEN EXTFILE:=TRUE;%
END;%

IF EXTFILE AND NOT PARAM THEN%
BEGIN%
  IF NUMFILES GEQ MAXFILES THEN ERROR(97)%
  ELSE%
    NUMFILES := NUMFILES + 1;%
    FILETAB[NUMFILES] := "NAM = 1"%
    GEN("DEFINE",7,2); GENID("F",LEVEL1000+NAM,5);%

    GEN("=",1,7);%

    GEN(FNAME,FNLENGTH,FNSTART); %

    GEN("#",2,6); GEN("SAVE",5,4); GEN("FILE",5,4);%

    GEN(FNAME,FNLENGTH,FNSTART); %
  END ELSE%
BEGIN%
  GEN("FILE",5,4); GENID("F",LEVEL1000+NAM,5);%
END;%

IF NOT PARAM THEN%
BEGIN%
  GEN("DISK",6,3); GEN("SERIAL",7,2);%

  IF EXTFILE THEN%
  BEGIN%
    IF ALGOLCNT LSS 13 THEN WRITEALGOL;%
    GEN("[010]",5,3);%

    GEN("",1,7);%

    GEN(FNAME,FNLENGTH,FNSTART); %

    GEN("",1,7); GEN("/ ",1,7);%

    IF ALGOLCNT<9 THEN WRITEALGOL;%
    GEN("",1,7); GEN(USER,7,1); GEN("",1,7);%

  END ELSE%
BEGIN%
  GEN("[20]",4,4); GEN("300]",4,4);%

```

```

03004 T 0318
03005 T 0320
03006 T 0324
00003008
03007 T 0324
03008 T 0325
03009 T 0326
03010 T 0327
03011 T 0328
03012 T 0329
03013 T 0331
00003019
00003019
03014 T 0353
00003014
03015 T 0362
00003015
03016 T 0371
00003016
00003016
00003016
03017 T 0398
00003017
03018 T 0407
00003008
03019 T 0407
03020 T 0411
00003020
00003020
03021 T 0432
00003019
03022 T 0432
03023 T 0433
03024 T 0433
00003024
00003024
03025 T 0451
03026 T 0451
03027 T 0452
03028 T 0454
00003028
03029 T 0463
00003029
03030 T 0472
00003030
03031 T 0481
00003031
00003031
03032 T 0499
03033 T 0500
00003033
00003033
00003033
03034 T 0532
00003026
03035 T 0532
03036 T 0532

```

```

END;%
GEN("(",3,5));%
RECSIZE:=IF T1,FORM=TEXTFILE THEN 10 ELSE%
          IF TYPETAB1[T1,FILETYPE],STRUCT=0 THEN 1 ELSE%
          TYPETAB3[T1,FILETYPE]-TYPETAB2[T1,FILETYPE]+1;%
GENINT(RECSIZE); GEN(")",1,7));%

IF RECSIZE=1 OR RECSIZE=10 THEN GENINT(150);%

ELSE GENINT(RECSIZE));%

IF ALGOLCNT LSS 10 THEN WRITEALGOL;%
GEN(",SAVE",6,3); GEN("30",2,6));%

GEN(")",2,6));%
END ELSE GEN(")",1,7));%

GEN("ARRAY",6,3); GENID("V",LEVEL1000+NAM,5));%

GEN("[",1,7));%

IF TYPETAB1[T1,FILETYPE],STRUCT=0 THEN%
BEGIN%
IF PARAM THEN GEN("0",1,7) ELSE GEN("0:0",3,5));%

END ELSE%

BEGIN%
GENINT(TYPETAB2[T1,FILETYPE]);%

IF NOT PARAM THEN%
BEGIN GEN(")",1,7); GENINT(TYPETAB3[T1,FILETYPE]) END;%

END;%

GEN(")",2,6));%

GEN("INTEGER",8,1); GENID("I",LEVEL1000+NAM,5));%

```

```

00003036
00003036
03037 T 0550 00003035
03038 T 0550 00003038
03039 T 0559
03040 T 0562
03041 T 0565
03042 T 0568
00003042
00003042
00003042
03043 T 0624 00003048
00003049
03044 T 0670 00003044
00003044
03045 T 0714
03046 T 0716 00003046
00003046
03047 T 0734 00003047
03048 T 0743 00003029
00003048
03049 T 0755 00003049
00003049
03050 T 0776 00003050
03051 T 0785
03052 T 0787
03053 T 0788 00003053
00003053
03054 T 0809 00003052
03055 T 0809
03056 T 0809 00003056
00003056
03057 T 0854
03058 T 0854 00003058
00003058
00003058
00003058
03059 T 0909 00003055
03060 T 0909 00003060
03061 T 0918 00003061
00003061
00003061

```

```

        GEN(";",1,7);%
    END;%
END;%
END OF LOOP;%
IF REALVAR OR ARRAYVAR THEN GEN(";",1,7);%
END OF DECLAREVARS;%

```

```

03062 T 0939
00003062
03063 T 0948
00002990
03064 T 0948
00002972
03065 T 0948
00002951
03066 T 0951
00003066
03067 T 0961
00002943

```

38 IS 978 LONG, NEXT SEG 2

```

%
%
PROCEDURE PARAMETERLIST;%
BEGIN%
    INTEGER FIRSTPARAM,CURKIND,P1,PX,I,T;%
    BOOLEAN FIRST;%
%
    DEFINE NEWPARAM=%
    BEGIN%
        IF NUMPARAMS>=MAXPARAMS THEN%
            BEGIN ERROR(70); NUMPARAMS:=MAXPARAMS-10 END;%
        NUMPARAMS:=NUMPARAMS+1;%
    END OF NEWPARAM;%
%
    NEWPARAM; FIRSTPARAM:=NUMPARAMS;%

    IF CURSY=LPAR THEN%
    BEGIN%
        DO BEGIN%
            INSYMBOL;%
            IF CURSY=VARSY OR CURSY=FUNCSY OR CURSY=PROCSY THEN%
            BEGIN%
                CURKIND:=IF CURSY=VARSY THEN VAR ELSE%
                    IF CURSY=FUNCSY THEN FUNC ELSE PROC;%
                INSYMBOL;%
            END ELSE CURKIND:=CONST;%

            FIRST:=TRUE; P1:=NUMPARAMS+1;%
            DO BEGIN%
                IF FIRST THEN FIRST:=FALSE ELSE INSYMBOL;%
                IF CURSY=IDENTIFIER THEN%
                BEGIN%
                    NEWNAME(CURNAME1,CURNAME2,CURLEVEL+1);%

                    PX:=THISINDEX; PX.PARAMKIND:=CURKIND;%
                    PX.PARAMLEVEL:=CURLEVEL+1;%
                    NEWPARAM; PARAMTAB[NUMPARAMS]:=PX;%
                END%
            END%
        END%
    END%

```

```

03068 T 0375
03069 T 0375
03070 T 0375
03071 T 0375
03072 T 0375
START OF SEGMENT ***** 39
03073 T 0000
03074 T 0000
03075 T 0000
03076 T 0000
03077 T 0000
03078 T 0000
03079 T 0000
03080 T 0000
03081 T 0000
03082 T 0000
00003082
00003082
03083 T 0005
03084 T 0006
03085 T 0006
03086 T 0007
03087 T 0007
03088 T 0010
03089 T 0010
03090 T 0012
03091 T 0015
03092 T 0016
00003088
03093 T 0017
03094 T 0019
03095 T 0020
03096 T 0022
03097 T 0023
03098 T 0023
00003098
00003098
03099 T 0049
03100 T 0051
03101 T 0053

```

```

END ELSE ERROR(9));%
INSYMBOL;%
END UNTIL CURSY#COMMA;%

IF CURSY#COLON THEN%
BEGIN%
IF CURKIND#PROC THEN ERROR(90));%
INSYMBOL;%
IF CURSY#IDENTIFIER THEN%
BEGIN%
SEARCH;%

IF FOUND THEN%
BEGIN%
IF THISID.IDCLASS#TYPES THEN%
BEGIN%
T3:=THISID.TYPE;%
FOR I:=P1 STEP 1 UNTIL NUMPARAMS DO%
PARAMTAB[I].PARAMTYPE:=T3;%
IF CURKIND#CONST OR CURKIND#VAR THEN%
BEGIN%
T:=TYPETAB1[T3];%
IF T.FORM#FILES THEN%
FOR I:=P1 STEP 1 UNTIL NUMPARAMS DO%
PARAMTAB[I].PARAMFILE:=1;%
IF T.STRUCT>0 AND CURKIND#CONST THEN ERROR(94));%
END ELSE IF T.STRUCT>0 THEN ERROR(38));%

END ELSE BEGIN ERROR(7); T3:=0 END;%

END ELSE BEGIN ERROR(1); T3:=0 END;%

END ELSE BEGIN ERROR(9); T3:=0 END;%

INSYMBOL;%
END ELSE%

BEGIN%
IF CURKIND#PROC THEN ERROR(7));%
T3:=0;%
END;%

T3.IDCLASS:=CURKIND; T3.FORMAL:=1;%
FOR I:=P1 STEP 1 UNTIL NUMPARAMS DO%
NAMETAB3[CURLEVEL+1,PARAMTAB[I].PARAMNAME]:=T3;%
END UNTIL CURSY#SEMICOLON;%

IF CURSY#RPAR THEN%

```

```

00003101
00003101
03102 T 0059 00003097
03103 T 0060
03104 T 0061 00003094
03105 T 0062
03106 T 0063
03107 T 0063
03108 T 0065
03109 T 0066
03110 T 0067
03111 T 0067
00003111
00003111
00003111
00003111
03112 T 0096
03113 T 0096
03114 T 0098
03115 T 0098
03116 T 0098
03117 T 0099
03118 T 0101
03119 T 0105
03120 T 0107
03121 T 0108
03122 T 0109
03123 T 0110
03124 T 0112
03125 T 0116
03126 T 0120
00003120
03127 T 0123 00003115
00003127
03128 T 0125 00003113
00003128
03129 T 0127 00003110
00003129
03130 T 0129
03131 T 0129 00003106
03132 T 0129
03133 T 0130
03134 T 0132
03135 T 0133 00003132
03136 T 0133
03137 T 0136
03138 T 0138
03139 T 0143 00003083
03140 T 0144

```

```

    BEGIN ERROR(46); SKIP(RPAR);%
      IF CURSY=RPAR THEN INSYMBOL;%
    END ELSE INSYMBOL;%

  END;%

  PARAMTAB[FIRSTPARAM]:=NUMPARAMS-FIRSTPARAM;%
END OF PARAMETERLIST;%

```

```

03141 T 0145
03142 T 0147
03143 T 0149
00003141
03144 T 0150
00003084
03145 T 0150
03146 T 0152
00003071
39 IS 157 LONG, NEXT SEG 2

```

```

%
%
PROCEDURE BLOCK;%
BEGIN%
  INTEGER INDEX,CTYPE,NUMFORWARDS,T,T3,IX,I;%

  REAL CVAL;%
  ALPHA C1,C2;%
  BOOLEAN VALUEPARAMS,FUN;%
  LABEL START;%

%
  INTEGER LABTABTOP,CONSTTABTOP,TYPETABTOP,PARAMTABTOP,TOPREC,%
  FORMERFIRSTLAB,FIRSTFILE;%

%
  FORMERFIRSTLAB:=FIRSTLAB;%
  LABTABTOP:=NUMLABS; FIRSTLAB:=LABTABTOP+1;%
  CONSTTABTOP:=NUMCONSTS;%
  TYPETABTOP:=NUMTYPES;%
  PARAMTABTOP:=NUMPARAMS;%
  TOPREC:=LASTREC;%
  FIRSTFILE:=NUMFILES+1;%

%
  TOPLEVEL:=CURLEVEL;%
  IF CURLEVEL>1 THEN GEN("BEGIN",6,3);%

START;%
  IF CURSY=LABELSY THEN
    BEGIN
      GEN("LABEL",6,3);%

      DO BEGIN%
        INSYMBOL;%
        IF CURSY=INTCONST THEN%
          BEGIN%
            GENID("L",CURVAL,4);%

            IF CURVAL>9999 THEN ERROR(33);%
            FOR I:=FIRSTLAB STEP 1 UNTIL NUMLABS DO%
              IF LABTAB[I].LABVAL=CURVAL THEN ERROR(31);%
              IF NUMLABS<=MAXLABS THEN BEGIN ERROR(34); NUMLABS:=0 END;%

            NUMLABS:=NUMLABS+1;%
            LABTAB[NUMLABS]:=CURVAL;%

```

```

03147 T 0375
03148 T 0375
03149 T 0375
03150 T 0375
03151 T 0375
START OF SEGMENT ***** 40
03152 T 0000
03153 T 0000
03154 T 0000
03155 T 0000
03156 T 0000
03157 T 0000
03158 T 0000
03159 T 0000
03160 T 0000
03161 T 0000
03162 T 0002
03163 T 0003
03164 T 0004
03165 T 0005
03166 T 0005
03167 T 0007
03168 T 0007
03169 T 0007
00003169
03170 T 0018
03171 T 0018
03172 T 0018
03173 T 0019
00003173
03174 T 0028
03175 T 0029
03176 T 0029
03177 T 0030
03178 T 0030
00003178
03179 T 0042
03180 T 0044
03181 T 0049
03182 T 0054
00003182
03183 T 0056
03184 T 0058

```

```

      INSYMBOL;%
      END ELSE BEGIN ERROR(10); SKIP(COMMA) END;%

      IF CURSY=COMMA THEN GEN(", ",1,7);%
      END UNTIL CURSY≠COMMA;%
      IF CURSY≠SEMICOLON THEN BEGIN ERROR(25); SKIP(SEMICOLON) END;%
      GEN("; ",1,7);%
      IF SYMKIND[CURSY]≠INITIAL THEN INSYMBOL;%
      END OF LABEL DECLARATION;%

```

x

```

      IF CURSY=CONSTSY THEN
      BEGIN
      INSYMBOL;%
      DO BEGIN%
      IF CURSY=IDENTIFIER THEN%
      BEGIN%
      NEWNAME(CURNAME1,CURNAME2,CURLEVEL); INDEX:=THISINDEX;%

      INSYMBOL;%
      IF CURSY=EQLSY THEN%
      BEGIN%
      INSYMBOL; CONSTANT(CVAL,CTYPE);%

```

```

      T3:=CTYPE; T3.IDCLASS:=CONST;%
      IF CVAL.[46:8]≠0 OR CVAL>1023 THEN%
      BEGIN%
      IF NUMCONSTS≥MAXCONSTS THEN%
      BEGIN ERROR(35); NUMCONSTS:=0 END;%

      NUMCONSTS:=NUMCONSTS+1;%
      CONSTTAB[ NUMCONSTS ]:=CVAL;%
      T3.INFO:=1023+NUMCONSTS;%

```

```

03185 T 0059
03186 T 0059
00003177
00003186
03187 T 0061
00003187
03188 T 0072
00003174
03189 T 0073
00003189
03190 T 0076
00003190
03191 T 0085
03192 T 0087
00003172
03193 T 0087
03194 T 0087
03195 T 0087
03196 T 0088
03197 T 0088
03198 T 0089
03199 T 0089
03200 T 0090
00003200
00003200
03201 T 0113
03202 T 0113
03203 T 0114
03204 T 0115
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
00003204
03205 T 0203
03206 T 0205
03207 T 0207
03208 T 0208
03209 T 0209
00003209
03210 T 0211
03211 T 0212
03212 T 0213

```



```

        END ELSE T3.INFO:=CVAL;%
        NAMETAB3[CURLEVEL,INDEX]:=T3;%
        END ELSE BEGIN ERROR(36); SKIP(SEMICOLON) END;%

        END ELSE BEGIN ERROR(9); SKIP(SEMICOLON) END;%

        IF CURSY#SEMICOLON THEN BEGIN ERROR(25); SKIP(SEMICOLON) END;%
        IF SYMKIND[CURSY]#INITIAL THEN INSYMBOL;%
        END UNTIL CURSY#IDENTIFIER;%

    END OF CONSTANT DECLARATION;%

```

*

```

    IF CURSY=TYPE SY THEN
    BEGIN
        INSYMBOL;%
        DO BEGIN%
            IF CURSY=IDENTIFIER THEN%
            BEGIN%
                NEWNAME(CURNAME1,CURNAME2,CURLEVEL); INDEX:=THISINDEX;%
            END%
        END%

        INSYMBOL;%
        IF CURSY=EQLSY THEN%
        BEGIN%
            INSYMBOL;%
            TYPEDECL(CTYPE, TX);%
            T3:=CTYPE; T3.IDCLASS:=TYPES;%
            NAMETAB3[CURLEVEL,INDEX]:=T3;%
            END ELSE BEGIN ERROR(36); SKIP(SEMICOLON) END;%

        END ELSE BEGIN ERROR(9); SKIP(SEMICOLON) END;%

        IF CURSY#SEMICOLON THEN BEGIN ERROR(25); SKIP(SEMICOLON) END;%
        IF SYMKIND[CURSY]#INITIAL THEN INSYMBOL;%
        END UNTIL CURSY#IDENTIFIER;%

    END OF TYPE DECLARATION;%

```

*

```

    IF CURSY=VAR SY THEN
    BEGIN
        VARINDEX:=0;%
        DO BEGIN%
            FIRSTVAR:=VARINDEX+1;%
            DO BEGIN%
                IF CURSY=VAR SY OR CURSY=COMMA THEN INSYMBOL;%
                IF CURSY=IDENTIFIER THEN%
                BEGIN%
                    IF VARINDEX>LISTLENGTH THEN%

```

```

03213 T 0215
00003207
03214 T 0218
03215 T 0220
00003203
00003215
03216 T 0222
00003199
00003216
03217 T 0224
00003217
03218 T 0226
03219 T 0228
00003197
03220 T 0230
00003195
03221 T 0230
03222 T 0230
03223 T 0230
03224 T 0231
03225 T 0231
03226 T 0232
03227 T 0232
03228 T 0233
00003228
00003228
03229 T 0256
03230 T 0256
03231 T 0257
03232 T 0258
03233 T 0258
03234 T 0259
03235 T 0262
03236 T 0264
00003231
0000323A
03237 T 0266
00003227
00003237
03238 T 0268
00003238
03239 T 0270
03240 T 0272
00003225
03241 T 0274
00003223
03242 T 0274
03243 T 0274
03244 T 0274
03245 T 0275
03246 T 0276
03247 T 0276
03248 T 0277
03249 T 0278
03250 T 0280
03251 T 0281
03252 T 0282

```

```

      BEGIN ERROR(37); VARINDEX:=0 END;%

      VARINDEX:=VARINDEX+1;%
      NEWNAME(CURNAME1,CURNAME2,CURLEVEL);%

      VARLIST[VARINDEX]:=THISINDEX;%
      INSYMBOL;%
      END ELSE BEGIN ERROR(9); SKIP(COLON) END;%

      END UNTIL CURSY#COMMA;%

      IF CURSY#COLON THEN BEGIN ERROR(26); SKIP(COLON) END;%

      IF CURSY=COLON THEN%
      BEGIN%
        INSYMBOL;%
        TYPEDECL(CTYPE,TX);%
        T3:=CTYPE; T3.IDCLASS:=VAR;%
        FOR I:=FIRSTVAR STEP 1 UNTIL VARINDEX DO%
          NAMETAB3[CURLEVEL,VARLIST[I]]:=T3;%
        END ELSE BEGIN ERROR(26); SKIP(SEMICOLON) END;%

        IF CURSY#SEMICOLON THEN BEGIN ERROR(25); SKIP(SEMICOLON) END;%

        IF SYMKIND[CURSY]#INITIAL THEN INSYMBOL;%
        END UNTIL CURSY#IDENTIFIER;%

        DECLAREVARS(FALSE,VARLIST,1,VARINDEX,CURLEVEL);%
        END OF VARIABLE DECLARATION;%

%
      IF NumpNtrs>0 THEN%
      BEGIN%
        C1:=CURNAME1; C2:=CURNAME2;%
        FOR I:=1 STEP 1 UNTIL NumpNtrs DO%
          BEGIN%
            CURNAME1:=PNTRTAB1[I]; CURNAME2:=PNTRTAB2[I];%
            SEARCHTAB(CURLEVEL);%

            THISID:=NAMETAB3[CURLEVEL,THISINDEX];%
            IF FOUND AND THISID.IDCLASS#TYPES THEN%
              TYPETAB1[PNTRTAB3[I]],POINTTYPE:=THISID.TYPE ELSE ERROR(62);%
            END;%

            CURNAME1:=C1; CURNAME2:=C2; NumpNtrs:=0;%
          END;%

%
      WHILE CURSY#FUNCSY OR CURSY#PROCSY DO %*** PROC/FUNC DECLARATION ***
      BEGIN
        FUN:=CURSY#FUNCSY; INSYMBOL;%
        IF CURSY#IDENTIFIER THEN%
          BEGIN%

```

```

03253 T 0282
00003253
03254 T 0284
03255 T 0286
00003255
00003255
03256 T 0308
03257 T 0309
03258 T 0310
00003251
00003258
03259 T 0312
00003248
03260 T 0313
00003260
03261 T 0316
03262 T 0317
03263 T 0317
03264 T 0318
03265 T 0319
03266 T 0321
03267 T 0323
03268 T 0327
00003262
00003268
03269 T 0329
00003269
03270 T 0332
03271 T 0334
00003246
03272 T 0335
03273 T 0337
00003244
03274 T 0337
03275 T 0337
03276 T 0338
03277 T 0339
03278 T 0340
03279 T 0342
03280 T 0342
03281 T 0344
00003281
00003281
03282 T 0362
03283 T 0364
03284 T 0366
03285 T 0371
00003279
03286 T 0373
03287 T 0375
00003276
03288 T 0375
03289 T 0375
03290 T 0378
03291 T 0378
03292 T 0380
03293 T 0380

```

```

SEARCHTAB(CURLEVEL);%

THISID:=NAMETAB3(CURLEVEL,THISINDEX);%
IF FOUND AND THISID.IDCLASS>=PROC THEN%
BEGIN%
  INDEX:=THISINDEX;%
  IF THISID.FORWARDDEF=1 THEN%
  BEGIN%
    NAMETAB3(THISLEVEL,THISINDEX).FORWARDDEF:=0;%
    NUMFORWARDS:=NUMFORWARDS-1;%
    IF(THISID.IDCLASS=PROC AND FUN)OR%
      (THISID.IDCLASS=FUNC AND NOT FUN) THEN ERROR(43);%
    INSYMBOL;%
  END ELSE BEGIN ERROR(2); SKIP(SEMICOLON) END;%

END ELSE%

BEGIN%
  NEWNAME(CURNAME1,CURNAME2,CURLEVEL); INDEX:=THISINDEX;%

  T3:=0; T3.INFO:=NUMPARAMS+1;%
  T3.IDCLASS:=IF FUN THEN FUNC ELSE PROC;%
  NAMETAB3(CURLEVEL,INDEX):=T3;%
  INSYMBOL; PARAMETERLIST;%
  IF CURSY=COLON THEN%
  BEGIN%
    IF NOT FUN THEN ERROR(48);%
    INSYMBOL;%
    IF CURSY=IDENTIFIER THEN%
    BEGIN%
      SEARCH;%

      IF FOUND THEN%
      BEGIN%
        IF THISID.IDCLASS=TYPES THEN%
        BEGIN%
          T:=TYPETAB1(THISID,TYPE);%
          IF T.FORM<=ALFA OR T.FORM=POINTERS THEN%
          BEGIN%
            NAMETAB3(CURLEVEL,INDEX).TYPE:=THISID.TYPE;%
          END ELSE ERROR(38);%

          END ELSE ERROR(7);%

        END ELSE ERROR(1);%

      END ELSE ERROR(9);%

      INSYMBOL;%
    END ELSE IF FUN THEN%

```

```

03294 T 0381
00003294
00003294
03295 T 0399
03296 T 0401
03297 T 0403
03298 T 0403
03299 T 0404
03300 T 0405
03301 T 0406
03302 T 0409
03303 T 0410
03304 T 0412
03305 T 0415
03306 T 0416
00003300
00003306
03307 T 0418
00003297
03308 T 0418
03309 T 0418
00003309
00003309
03310 T 0442
03311 T 0445
03312 T 0448
03313 T 0450
03314 T 0451
03315 T 0452
03316 T 0452
03317 T 0454
03318 T 0455
03319 T 0455
03320 T 0456
00003320
00003320
00003320
00003320
03321 T 0485
03322 T 0485
03323 T 0485
03324 T 0487
03325 T 0487
03326 T 0489
03327 T 0491
03328 T 0492
03329 T 0496
00003327
03330 T 0497
00003324
03331 T 0498
00003328
03332 T 0499
00003319
03333 T 0501
03334 T 0501
00003315

```

```

        BEGIN ERROR(26); SKIP(SEMICOLON) END;%
    END;%
END ELSE BEGIN ERROR(9); SKIP(SEMICOLON) END;%

IF CURSY≠SEMICOLON THEN BEGIN ERROR(25); SKIP(SEMICOLON) END;%
IF FUN THEN GEN("FUNCTN",7,2) ELSE%
GEN("PROCEDU",8,1); GENID("V",1000×CURLEVEL+INDEX,5);%

T:=NAMETAB3(CURLEVEL,INDEX).INFO; TX:=T+PARAMTAB[T];%
IF TX>T THEN%
BEGIN%
    GEN(", ",1,7);%

    FOR I:=T+1 STEP 1 UNTIL TX DO%
    BEGIN GENID("V",1000×(CURLEVEL+1)+PARAMTAB[I].PARAMNAME,5);%

        IF BOOLEAN(PARAMTAB[I].PARAMFILE) THEN%
        BEGIN%
            GEN(", ",1,7);%

            GENID("F",1000×(CURLEVEL+1)+PARAMTAB[I].PARAMNAME,5);%

            GEN(", ",1,7);%

            GENID("I",1000×(CURLEVEL+1)+PARAMTAB[I].PARAMNAME,5);%

        END;%

        IF I LSS TX THEN GEN(", ",1,7);%

    END;%

    GEN(", ",2,6);%

    VALUEPARAMS:=FALSE;%
    FOR I:=T+1 STEP 1 UNTIL TX DO%
    IF PARAMTAB[I].PARAMKIND=CONST THEN%
    BEGIN%
        IF NOT VALUEPARAMS THEN%
        BEGIN GEN("VALUE",6,3);%

            VALUEPARAMS:=TRUE;%
        END ELSE GEN(", ",1,7);%

        GENID("V",1000×(CURLEVEL+1)+PARAMTAB[I].PARAMNAME,5);%

    END;%

    IF VALUEPARAMS THEN GEN(", ",1,7);%

```

```

03335 T 0502
00003335
03336 T 0504
00003308
03337 T 0504
00003293
00003337
03338 T 0506
00003338
03339 T 0509
00003339
03340 T 0518
00003340
00003340
03341 T 0543
03342 T 0546
03343 T 0547
03344 T 0548
00003344
03345 T 0557
03346 T 0561
00003346
03347 T 0575
03348 T 0576
03349 T 0577
00003349
03350 T 0586
00003350
03351 T 0600
00003351
03352 T 0609
00003352
03353 T 0625
00003348
03354 T 0625
00003354
03355 T 0635
00003346
03356 T 0635
00003356
03357 T 0644
03358 T 0645
03359 T 0649
03360 T 0651
03361 T 0651
03362 T 0652
00003362
03363 T 0661
03364 T 0662
00003362
00003364
03365 T 0674
00003362
03366 T 0688
00003360
03367 T 0688
00003367

```

```
DECLAREVARS(TRUE,PARAMTAB,T+1,TX,CURLEVEL+1);%
END ELSE GEN(";",1,7);%
```

```
03368 T 0698
03369 T 0701
00003343
00003369
```

%

```
INSYMBOL;%
IF CURNAME1="7FORWAR" AND CURNAME2="D" THEN%
BEGIN%
NAMETAB3[CURLEVEL,INDEX],FORWARDDEF:=1;%
NUMFORWARDS:=NUMFORWARDS+1;%
GEN("FORWARD",8,1);%
```

```
03370 T 0711
03371 T 0711
03372 T 0711
03373 T 0713
03374 T 0714
03375 T 0719
03376 T 0718
```

```
INSYMBOL;%
END ELSE%
```

```
00003376
03377 T 0727
03378 T 0728
```

```
BEGIN%
CURLEVEL:=CURLEVEL+1;%
IF CURLEVEL>=LASTREC THEN ERROR(55);%
BLOCKTAB[CURLEVEL]:=NUMBLOCKS:=NUMBLOCKS+1;%
T:=CURFUNC; CURFUNC:=IF FUN THEN INDEX ELSE -1;%
BLOCK; %*** COMPILER PROCEDURE BODY ***
REPLACE POINTER(NAMETAB1[CURLEVEL,*]) BY 0%
FOR MAXNAMES+1 WORDS;%
CURLEVEL:=CURLEVEL-1; CURFUNC:=T;%
TOPLEVEL:=CURLEVEL;%
END;%
```

```
00003373
03379 T 0728
03380 T 0731
03381 T 0732
03382 T 0734
03383 T 0736
03384 T 0739
03385 T 0740
03386 T 0742
03387 T 0746
03388 T 0748
03389 T 0749
```

```
IF CURSY≠SEMICOLON THEN BEGIN ERROR(25); SKIP(SEMICOLON) END;%
```

```
00003379
03390 T 0749
```

```
GEN(";",1,7);%
```

```
00003390
```

```
IF SYMKIND[CURSY]≠INITIAL THEN INSYMBOL;%
END OF PROCEDURE DECLARATION;%
```

```
03391 T 0751
00003391
```

%
%

```
IF NUMFORWARDS>0 THEN ERROR(44);%
GEN("INTEGER",8,1);%
```

```
03392 T 0760
03393 T 0762
```

```
FOR I:=1 STEP 1 UNTIL MAXTEMPS DO%
BEGIN GENID("T",I,2);%
```

```
00003290
03394 T 0763
03395 T 0763
```

```
IF I<MAXTEMPS THEN GEN(";",1,7) ELSE GEN(";",1,7);%
```

```
03396 T 0763
03397 T 0766
```

```
END;%
```

```
00003397
03398 T 0775
03399 T 0777
```

```
IF CURSY≠BEGINSY THEN%
BEGIN ERROR(39);%
WHILE SYMKIND[CURSY]≠INITIAL DO%
BEGIN INSYMBOL; SKIP(SEMICOLON) END;%
```

```
00003399
03400 T 0789
```

```
IF(CURSY=TYPE$Y)OR(CONSTSY≤CURSY AND CURSY≤PROCSY)THEN%
GO TO START;%
```

```
00003400
03401 T 0808
```

```
END;%
```

```
00003399
03402 T 0811
03403 T 0811
```

```
IF CURLEVEL=1 THEN%
```

```
03404 T 0813
03405 T 0814
```

```
00003405
03406 T 0816
03407 T 0819
```

```
03408 T 0819
```

```
00003403
03409 T 0819
```

```

BEGIN%
  GEN("INIT(",5,3));%
  IF INPUTDECL THEN GEN("TRUE",4,4) ELSE GEN("FALSE",5,3));%
  GEN(")",2,6));%
END;%
FOR I:=FIRSTFILE STEP 1 UNTIL NUMFILES DO%
  IF FILETAB[I] LSS 0 THEN%
    BEGIN%
      GEN("CHFIL(",6,2));%
      GENID("F",1000*CURLEVEL+FILETAB[I]-1,5));%
      GEN(")",2,6));%
    END%
  ELSE%
    BEGIN%
      GENID("I",1000*CURLEVEL+FILETAB[I],5));%
      GEN(", ",1,7); GEN("BUFSIZE",7,1); GEN("I=80;",5,3));%
    END;%
  NUMFILES:=FIRSTFILE-1;%
  COMPSTAT;   %*** COMPILE STATEMENT PART ***
  FOR I:=LASTREC STEP 1 UNTIL TOPREC=1 DO%
    REPLACE POINTER(NAMETAB1[I,*]) BY 0 FOR MAXNAMES+1 WORDS;%
    FOR J:=FIRSTLAB STEP 1 UNTIL NUMLABS DO%
      IF LABTAB[J].LABDEF=0 THEN ERROR(93));%
    LASTREC:=TOPREC;%
    NUMLABS:=LABTABTOP;%
    FIRSTLAB:=FORMERFIRSTLAB;%
    NUMCONSTS:=CONSTTABTOP;%
    NUMTYPES:=TYPETABTOP;%
    NUMPARAMS:=PARAMTABTOP;%
    IF CURLEVEL>1 THEN GEN("END",4,5));%
  END OF BLOCK;%

```

```

03410 T 0820
03411 T 0821
03412 T 0830
03413 T 0852
03414 T 0861
03415 T 0861
03416 T 0864
03417 T 0865
03418 T 0865
03419 T 0874
03420 T 0888
03421 T 0897
03422 T 0897
03423 T 0897
03424 T 0900
03425 T 0913
03426 T 0940
03427 T 0942
03428 T 0943
03429 T 0943
03430 T 0944
03431 T 0944
03432 T 0948
03433 T 0957
03434 T 0958
03435 T 0963
03436 T 0963
03437 T 0964
03438 T 0965
03439 T 0966
03440 T 0966
03441 T 0967

```

```

00003411
00003412
00003412
00003413
00003410
00003418
00003419
00003420
00003417
00003424
00003425
00003425
00003425
00003423
00003441
00003150

```

40 IS 988 LONG, NEXT SEG 2

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
%
%
%
%
%
%
%
%
%
%
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
%
%
INTEGER PROGLNAMELENGTH;%
ALPHA PROGLNAME,ALGOLNAME;%
%
ALGOLNAME:="PASC000"&ENTIER(TIME(4)MOD 10)[17:5:6];%
ALGOLNAME:=ALGOLNAME&ENTIER(TIME(4)DIV 7)[11:5:6];%
ALGOLNAME:=ALGOLNAME&ENTIER(TIME(4)MOD 9)[5:5:6];%
USER:=TIME(-1);%
FILL PASCALGOL WITH ALGOLNAME,USER;%
BEGIN%
  FILE PASCALGOL DISK SERIAL "PASCALGOL"/"DISK" (2,10,150);%

  ARRAY BUF[0:9];%
  LABEL EOF;%

%
  WHILE TRUE DO%
  BEGIN%
    READ(PASCALGOL,9,BUF[*]) [EOF];%
    WRITE(PASCALGOL,10,BUF[*]);%
  END;%

EOF;%
END OF TRANSFER OF RUN TIME SYSTEM;%

CARDLENGTH:=72;%
INITIALIZE; NEWCARD;%

LISTOPTION:=CHECKOPTION:=TRUE;%
CI=# "; INSYMBOL;%
IF CURSY=PROGRAMSY THEN%
BEGIN%
  INSYMBOL;%
  IF CURSY=IDENTIFIER THEN%
  BEGIN%
    PROGLNAME:=CURNAME1.[35:36]; PROGLNAMELENGTH:=MIN(6,CURLENGTH);%
    INSYMBOL;%
    IF CURSY=LPAR THEN%
    BEGIN%
      DO BEGIN%
        INSYMBOL;%
        IF CURSY=IDENTIFIER THEN%
        BEGIN%

```

```

03444 T 0375
03445 T 0375
03446 T 0375
03447 T 0375
03448 T 0375
03449 T 0375
03450 T 0375
03451 T 0375
03452 T 0375
03453 T 0375
03454 T 0375
03455 T 0375
03456 T 0375
03457 T 0375
03458 T 0375
03459 T 0375
03460 T 0378
03461 T 0382
03462 T 0386
03463 T 0387
03464 T 0391
03465 T 0391
START OF SEGMENT ***** 41
03466 T 0004
03467 T 0006
03468 T 0006
03469 T 0006
03470 T 0007
03471 T 0007
03472 T 0013
03473 T 0017
00003470
03474 T 0017
03475 T 0018
00003464
41 IS 21 LONG, NEXT SEG 2
03476 T 0394
03477 T 0394
00003477
00003477
00003477
00003477
03478 T 0460
03479 T 0462
03480 T 0463
03481 T 0464
03482 T 0464
03483 T 0465
03484 T 0465
03485 T 0466
03486 T 0470
03487 T 0471
03488 T 0471
03489 T 0472
03490 T 0473
03491 T 0473
03492 T 0474

```

```

IF CURNAME1="50INPUT" THEN INPUTDECL:=TRUE ELSE%
IF CURNAME1="60OUTPUT" THEN OUTPUTDECL:=TRUE ELSE%
BEGIN%
  IF CURLLENGTH>6 THEN ERROR(77);%
  NUMEXTFILES1=NUMEXTFILES+1;%
  IF NUMEXTFILES<MAXEXTFILES THEN%
    EXTFILETAB[1]=CURNAME1 ELSE%
  IF NUMEXTFILES=MAXEXTFILES+1 THEN ERROR(73);%
END;%

END ELSE ERROR(9);%

  INSYMBOL;%
  END UNTIL CURSY#COMMA;%

  IF CURSY#RPAR THEN BEGIN ERROR(46); SKIP(SEMICOLON) END;%

  IF CURSY#RPAR THEN INSYMBOL;%
  IF CURSY#SEMICOLON THEN BEGIN ERROR(25); SKIP(SEMICOLON) END;%

END ELSE BEGIN ERROR(58); SKIP(SEMICOLON) END;%

END ELSE BEGIN ERROR(9); SKIP(SEMICOLON) END;%

END ELSE BEGIN ERROR(75); SKIP(SEMICOLON) END;%

INSYMBOL;%
CURLEVEL:=1;%
LASTREC:=MAXTABLES+1;%
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
%
      BLOCK;          %   COMPILER USER PROGRAM.          %
%
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
IF CURSY#DOT THEN%
BEGIN%
  ERROR(76);%
  DO BLOCK UNTIL CURSY=DOT;%
END;%

IF FALSE THEN%
BEGIN%
ENDOFINPUT: ERROR(87);  CHARCNT:=1;%
      WRITE(LINES,TERMMESS);%
END;%

IF LISTOPTION AND CHARCNT<0 THEN PRINTLINE;%

IF ERRINX>0 THEN PRINTERRORS;%
WRITE(LINES[DBL]);%
WRITE(LINES[DBL]);%
IF NUMERRS=0 THEN%

```

```

03493 T 0474
03494 T 0476
03495 T 0481
03496 T 0483
03497 T 0485
03498 T 0486
03499 T 0487
03500 T 0488
03501 T 0491
          00003495
03502 T 0491
          00003492
03503 T 0493
03504 T 0493
          00003489
03505 T 0494
          00003505
03506 T 0497
03507 T 0499
          00003507
03508 T 0502
          00003488
          00003508
03509 T 0504
          00003484
          00003509
03510 T 0506
          00003481
          00003510
03511 T 0508
03512 T 0508
03513 T 0509
03514 T 0510
03515 T 0510
03516 T 0510
03517 T 0511
03518 T 0511
03519 T 0511
03520 T 0511
03521 T 0512
03522 T 0513
03523 T 0514
          00003520
03524 T 0514
03525 T 0515
03526 T 0515
03527 T 0517
03528 T 0520
          00003525
03529 T 0520
          00003529
          00003529
          00003529
03530 T 0565
03531 T 0567
03532 T 0571
03533 T 0575

```



```

BEGIN%
  ARRAY ZIPARRAY[0:19], Z[0:0];%
  POINTER ZIPPNT;%
%
  DEFINE ZIPTEXT(TEXT,L)=%
  BEGIN%
    Z[0]=TEXT;%
    REPLACE ZIPPNT:ZIPPNT BY POINTER(Z[*])+(8-L) FOR L;%
  END;%
%
  PROCEDURE ZIPNUM(N); % TRANSFERS A NUMBER TO THE ZIP BUFFER.
  VALUE N; INTEGER N;%
  IF N<9 THEN ZIPTEXT(N,1) ELSE%
  BEGIN ZIPNUM(N DIV 10); ZIPTEXT(ENTIER(N MOD 10),1) END;%

%
  WRITEALGOL;%
  WRITE(PASCALGOL, LASTLINE);%
  LOCK(PASCALGOL, SAVE);%
  ZIPPNT:=POINTER(ZIPARRAY[*]);%
  REPLACE ZIPPNT BY " " FOR 20 WORDS;%
  WRITE(LINES, NOERRORS);%
  ZIPTEXT("CC ",3); ZIPTEXT("COMPILE",7);%

  ZIPTEXT(" ",1); ZIPTEXT(PROGNAME, PROGNAMLENGTH);%

  ZIPTEXT("/ ",1); ZIPTEXT(USER,7);%

  ZIPTEXT(" XALGOL",7); ZIPTEXT(" ",1);%

  IF SAVEFACTOR>0 THEN ZIPTEXT("LIBRARY",7);%
  IF SAVEFACTOR<0 THEN ZIPTEXT("SYNTAX",6);%

  ZIPTEXT(")",1);%

  ZIPTEXT("XALGOL",6); ZIPTEXT(" FILE",5);%

  ZIPTEXT(" CARD=",6); ZIPTEXT(ALGOLNAME,7);%

  ZIPTEXT("/ ",1); ZIPTEXT(USER,7);%

  ZIPTEXT(" SERIAL",7); ZIPTEXT(")",1);%

```

```

03534 T 0576
03535 T 0576
START OF SEGMENT ***** 42
03536 T 0005
03537 T 0005
03538 T 0005
03539 T 0005
03540 T 0005
03541 T 0005
03542 T 0005
03543 T 0005
03544 T 0005
03545 T 0005
03546 T 0005
00003546
03547 T 0014
00003547
00003547

03548 T 0027
03549 T 0027
03550 T 0027
03551 T 0030
03552 T 0032
03553 T 0033
03554 T 0037
03555 T 0040
00003555
00003555
03556 T 0055
00003556
00003556
03557 T 0069
00003557
00003557
03558 T 0084
00003558
00003558
03559 T 0098
00003559
03560 T 0107
00003560
03561 T 0115
00003561
03562 T 0123
00003562
00003562
03563 T 0137
00003563
00003563
03564 T 0161
00003564
00003564
03565 T 0175

```

```

IF SAVEFACTOR>0 THEN%
BEGIN%
  ZIPTEXT("SAVE=",5); ZIPNUM(SAVEFACTOR);%

  ZIPTEXT(";",1);%

END;%

ZIPTXT("END,",4);%

ZIP WITH ZIPARRAY[*];%
END OF COMPILER ZIP ELSE%

```

```

BEGIN%
  INTEGER I;%

  SWITCH FORMAT ERRORMESS1 ;=%

```

```

(" 0 *** COMPILER ERROR *** CONTACT THE COMPUTER CENTRE."),%
(" 1 IDENTIFIER NOT DEFINED."),%
(" 2 IDENTIFIER ALREADY DEFINED."),%
(" 3 WRONG NUMBER OF PARAMETERS."),%
(" 4 SYNTAX ERROR."),%
(" 5 VARIABLE NOT ACCESSIBLE (HARDWARE RESTRICTION)."),%
(" 6 STRINGS MAY NOT BE CONTINUED FROM ONE CARD TO ANOTHER."),%
(" 7 A TYPE EXPECTED."),%
(" 8 VARIABLE EXPECTED."),%
(" 9 IDENTIFIER EXPECTED."),%
(" 10 INTEGER CONSTANT EXPECTED."),%
(" 11 CONSTANT OF OTHER TYPE THAN EXPECTED."),%
(" 12 VARIABLE OF ILLEGAL TYPE."),%
(" 13 UNRECOGNIZABLE STATEMENT."),%
(" 14 CONSTANT TOO BIG OR TOO SMALL."),%
(" 15 UNDEFINED LABEL."),%
(" 16 FOR- AND CASE-STATEMENTS NESTED TOO DEEP."),%
(" 17 EXPRESSION IS OF WRONG TYPE."),%
(" 18 ""OF"" EXPECTED."),%
(" 19 ""DO"" EXPECTED."),%
(" 20 ""ELSE"" WITHOUT CORRESPONDING ""THEN""."),%
(" 21 ILLEGAL TERMINATION OF STATEMENT."),%
(" 22 ""UNTIL"" EXPECTED."),%
(" 23 ""TO""/""DOWNTO"" EXPECTED."),%
(" 24 ""END"" EXPECTED."),%
(" 25 "";"" EXPECTED."),%
(" 26 "":"" EXPECTED."),%
(" 27 ""THEN"" EXPECTED."),%
(" 28 "":="" EXPECTED."),%
(" 29 ONLY NUMBERS MAY BE SIGNED."),%
(" 30 TOO MANY CASES."),%
(" 31 LABEL USED MORE THAN ONCE."),%
(" 32 CONSTANT EXPECTED."),%
(" 33 LABEL NOT IN RANGE 0.,9999."),%
(" 34 TOO MANY LABELS DECLARED."),%
(" 35 TOO MANY CONSTANTS DECLARED.")%

```

```

00003565
00003565
03566 T 0190
03567 T 0191
03568 T 0191
00003568
03569 T 0199
00003569
03570 T 0206
00003567
03571 T 0206
00003571
03572 T 0214
03573 T 0215
00003534
42 IS 222 LONG, NEXT SEG 2
03574 T 0577
03575 T 0577
START OF SEGMENT ***** 43
03576 T 0000
START OF SEGMENT ***** 44
03577 T 0000
03578 T 0000
03579 T 0000
03580 T 0000
03581 T 0000
03582 T 0000
03583 T 0000
03584 T 0000
03585 T 0000
03586 T 0000
03587 T 0000
03588 T 0000
03589 T 0000
03590 T 0000
03591 T 0000
03592 T 0000
03593 T 0000
03594 T 0000
03595 T 0000
03596 T 0000
03597 T 0000
03598 T 0000
03599 T 0000
03600 T 0000
03601 T 0000
03602 T 0000
03603 T 0000
03604 T 0000
03605 T 0000
03606 T 0000
03607 T 0000
03608 T 0000
03609 T 0000
03610 T 0000
03611 T 0000
03612 T 0000

```

(" 36	""="" EXPECTED,")%&	03613	T	0000
(" 37	THE LIST IS TOO LONG,")%&	03614	T	0000
(" 38	INVALID TYPE FOR A FUNCTION,")%&	03615	T	0000
(" 39	""BEGIN"" EXPECTED,")%&	03616	T	0000
(" 40	TOO MANY IDENTIFIERS DECLARED,")%&	03617	T	0000
(" 41	ALFA CONSTANS MAY NOT BE LONGER THAN 7 CHARACTERS,")%&	03618	T	0000
(" 42	EXPRESSION IS NOT OF TYPE BOOLEAN,")%&	03619	T	0000
(" 43	NOT PROPER FORWARD DECLARATION,")%&	03620	T	0000
(" 44	UNSATISFIED FORWARD DECLARATION,")%&	03621	T	0000
(" 45	TOO MANY DIFFERENT TYPES DECLARED,")%&	03622	T	0000
(" 46	"""" EXPECTED,")%&	03623	T	0000
(" 47	""[""" EXPECTED,")%&	03624	T	0000
(" 48	A SIMPLE TYPE EXPECTED,")%&	03625	T	0000
(" 49	""ARRAY OF ARRAY"" AND ""ARRAY OF RECORD"" ILLEGAL,")%&	03626	T	0000
	" AS FILE TYPE,")%&	03627	T	0000
(" 50	""FILE OF FILE"" IS ILLEGAL,")%&	03628	T	0000
(" 51	SET BOUNDRY IS TOO BIG OR TOO SMALL,")%&	03629	T	0000
(" 52	TOO MANY UNDECLARED POINTERS,")%&	03630	T	0000
(" 53	""."" EXPECTED,")%&	03631	T	0000
(" 54	FIRST VALUE IS GREATER THAN SECOND VALUE,")%&	03632	T	0000
(" 55	TOO MANY RECORDS DECLARED AT ONE TIME,")%&	03633	T	0000
(" 56	THE RECORD CONTAINS MORE THAN 1023 WORDS,")%&	03634	T	0000
(" 57	FILES NOT ALLOWED IN RECORDS,")%&	03635	T	0000
(" 58	""(""" EXPECTED,")%&	03636	T	0000
(" 59	""]"" EXPECTED,")%&	03637	T	0000

44 IS 590 LONG, NEXT SEG 43

x

SWITCH FORMAT ERRORMESS2 :=%

(" 60	""ARRAY OF FILE"" NOT ALLOWED,")%&	03638	T	0000
(" 61	RANGE OF INDEX IS GREATER THAN 1023,")%&	03639	T	0000
(" 62	UNSATISFIED POINTER DECLARATION,")%&	03640	T	0000
(" 63	EXPRESSION IS TOO LONG,")%&	03641	T	0000
(" 64	ILLEGAL OPERATOR FOR THIS TYPE OF EXPRESSION,")%&	03642	T	0000
(" 65	INTEGER EXPRESSION EXPECTED,")%&	03643	T	0000
(" 66	A SET EXPECTED,")%&	03644	T	0000
(" 67	PARAMETER OF ILLEGAL TYPE,")%&	03645	T	0000
(" 68	PROCEDURES NOT ALLOWED IN THIS CONTEXT,")%&	03646	T	0000
(" 69	ILLEGAL USE OF THIS TYPE OF IDENTIFIER,")%&	03647	T	0000
(" 70	TOO MANY PARAMETERS DECLARED IN THE PROGRAM,")%&	03648	T	0000
(" 71	""ARRAY OF CHAR"" EXPECTED,")%&	03649	T	0000
(" 72	WRONG TYPE OF SET EXPRESSION,")%&	03650	T	0000
(" 73	TOO MANY EXTERNAL FILES,")%&	03651	T	0000
(" 74	ILLEGAL IDENTIFIER FOR EXTERNAL FILE,")%&	03652	T	0000
(" 75	""PROGRAM"" EXPECTED,")%&	03653	T	0000
(" 76	""."" EXPECTED AT END OF PROGRAM,")%&	03654	T	0000
(" 77	EXTERNAL FILE IDENTIFIER MAY NOT EXCEED 6 CHARACTERS,")%&	03655	T	0000
(" 78	ILLEGAL FILE PARAMETER,")%&	03656	T	0000
(" 79	ILLEGAL USE OF FILE HANDLING PROCEDURE,")%&	03657	T	0000
(" 80	TEXT=FILE EXPECTED,")%&	03658	T	0000
(" 81	POINTER VARIABLE EXPECTED,")%&	03659	T	0000
(" 82	ONLY VALUES OF TYPE REAL, INTEGER OR CHAR MAY BE READ,")%&	03660	T	0000
(" 83	VARIABLES IN RECORDS ILLEGAL IN THIS CONTEXT,")%&	03661	T	0000
(" 84	DISPLAY OVERFLOW,")%&	03662	T	0000
(" 85	READ AND WRITE MAY ONLY BE USED ON TEXT=FILES,")%&	03663	T	0000
(" 86	REFERENCED OBJECT IS TOO BIG,")%&	03664	T	0000
(" 87	END=OF=INPUT DISCOVERED,")%&	03665	T	0000
		03666	T	0000
		03667	T	0000

START OF SEGMENT ***** 45

```

(" 88 CHARACTER ARRAY EXPECTED."):,%
(" 89 "" "" EXPECTED."):,%
(" 90 PROCEDURES MAY NOT HAVE ANY TYPE."):,%
(" 91 PARAMETER OF WRONG KIND."):,%
(" 92 ONLY COMPLETE ARRAYS AND RECORDS MAY BE TRANSMITTED."):,%
(" 93 DECLARED LABEL NOT USED."):,%
(" 94 PARAMETERS OF THIS TYPE SHOULD NOT BE VALUE PARAMETERS."):,%
(" 95 ASSIGNMENT OF STRUCTURED VARIABLES NOT IMPLEMENTED."):,%
(" 96 INPUT/OUTPUT NOT DECLARED."):,%
(" 97 TOO MANY FILES IN USE."):,%
(" 98 RECORD IDENTIFIER EXPECTED."):,%
(" 99 UNRECOGNIZABLE ITEM."):,%
():,%

```

```

03668 T 0000
03669 T 0000
03670 T 0000
03671 T 0000
03672 T 0000
03673 T 0000
03674 T 0000
03675 T 0000
03676 T 0000
03677 T 0000
03678 T 0000
03679 T 0000
03680 T 0000

```

```
%
%
```

```

WRITE(LINES,ERRORS,NUMERRS);%
FOR I:=0 STEP 1 UNTIL 59 DO IF ERR[I] THEN%
WRITE(LINES,ERRORMESS1[I]);%
FOR I:=60 STEP 1 UNTIL 119 DO IF ERR[I] THEN%
WRITE(LINES,ERRORMESS2[I-60]);%
END OF ERROR MESSAGES;%

```

```

45 IS 430 LONG, NEXT SEG 43
03681 T 0000
03682 T 0000
03683 T 0000
03684 T 0007
03685 T 0008
03686 T 0015
03687 T 0016
03688 T 0023

```

```

IF XREFOPTION THEN%
BEGIN%
REPLACE POINTER(XREFLINE[*]) BY " " FOR 17 WORDS;%
HEADING;%

```

```

0000357#
43 IS 28 LONG, NEXT SEG 2
03689 T 0578
03690 T 0578
03691 T 0578
03692 T 0588

```

```

SORT(PRINTXREF,XREFFILE,0,XREFMAX,XREFCOMPARE,3,1000,6000);%
END;%

```

```

00003692
03693 T 0604
03694 T 0624

```

```
END OF COMPILER.%
```

```

00003690
03695 T 0624

```

```

00000027
2 IS 630 LONG, NEXT SEG 1
START OF SEGMENT ***** 46
46 IS 4 LONG, NEXT SEG 1
1 IS 2 LONG, NEXT SEG 0
62 IS 69 LONG, NEXT SEG 0

```

FPB

00010200	00000000	00006246	64512325	60042321	51240002	01000000	00000000	43314525	62606005	43314525	62000314
00000000	00000047	21622321	43271147	21622321	43274643	00041400	00000000	00006751	25262631	43106751	25262631
43250005	14472162	23516445	24316242	60606007	47216223	51644500	06140000	00000000	00246251	63016060	05246251
63010007	14000000	00000000	24625163	02606005	24625163	02001002	00000000	00000024	62516303	60600524	62516303
00110200	00000000	00002462	51630460	60052462	51630400	12020000	00000000	00246251	63056060	05246251	63050013
02000000	00000000	24625163	06606005	24625163	06001402	00000000	00000024	62516307	60600524	62516307	00000000
10636411	00000000	10546113	20000000	10457536	04000000	10373465	45000000	10111240	27620000	00013510	35564000
00116432	45121000	00222141	16345200	00326571	42036440	00434327	72446150	00543415	71157602	00654321	27413543
00767405	55316473	01110530	71060221	01212657	07274266	01315432	71153343	01420741	47406234	01525132	01307703
01632360	41571663	01741054	52130240	02051267	64556310	02163545	61711772	02310047	71627437	02412061	70175347

SEGDICT

00000000	00000000	00002200	00201126	00002401	16601100	20017600	05600001	00020500	15100003	00022301	43500015
00000000	07600007	00000000	10100012	00020001	36100051	20027500	01500050	00031300	75200103	00031400	63600132
00031500	76200203	00031601	42000150	00032000	27100224	00032100	32500233	00030200	66700243	00030300	17000264
00032301	40500270	00032501	71700325	00032700	11200322	00033100	36200367	20033200	00600366	00033400	72100416
00033600	53000436	00034000	35200452	00034300	54000462	00034500	13000476	00034700	22200501	00035100	64700506
00035300	21600525	00035700	32300532	00036101	27700550	00036300	04400600	00036500	13500602	00034200	36400606
00037501	41100617	00037400	62100651	00037701	72200702	00040100	23500743	00040201	73400751	00040700	02501012
00041400	33601013	00042000	03401077	20042101	11601034	20042200	65601060	20022400	00401125	10030000	00100020
10027700	00100017	10024200	00100001	10000500	00100002	10042700	00100004	10030100	00100006	10027300	00100021
10001400	00100022	10001500	00100023	10001600	00100024	10042600	00100027	10003400	00100150	10024000	00100040
10027100	00100041	10020600	00100042	20003300	10501127	00000000	00000000	43314525	62606005	43314525	62000314
00000000	00000047	21622321	43271147	21622321	43274643	00041400	00000000	00006751	25262631	43106751	25262631
43250005	14472162	23516445	24316242	60606007	47216223	51644500	06140000	00000000	00246251	63016060	05246251
63010007	14000000	00000000	24625163	02606005	24625163	02001002	00000000	00000024	62516303	60600524	62516303
00110200	00000000	00002462	51630460	60052462	51630400	12020000	00000000	00246251	63056060	05246251	63050013

PRT

00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	65406200	06200000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
65406600	06600000	65406700	06700000	65407000	07000000	00000000	00000000	00000000	00000000	00000000	00000000
67400100	00100000	00000000	00000000	65027200	00201163	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	66407600	07600000	65407200	07200000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
66400300	00300000	00000000	00000000	65027600	01001347	65400200	00200105	65024100	00200273	00000000	00000000
00000000	00000000	65400400	00400146	65407500	07500000	00000000	00000000	00000000	00000000	65020100	00200133
65021100	00200145	00000000	00000000	65021200	00200202	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	65400500	00501432	66405600	05600000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	65407300	07300000	64021400	00200323
65406100	06100000	65020200	00200333	65024300	00200363	65024400	00200365	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	65407400	07400000
66042400	00201004	65406500	06500000	00000000	00000000	66401100	01100000	66401000	01000640	65406000	06000000
65405700	05700000	65406400	06400000	65402000	02000661	65402100	02100000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	65401200	01200746

65401300	01300630	65031700	01400755	66401500	01501403	66401400	01400114	65401600	01600266	65401700	01700322
00000000	00000000	65402200	02201370	00000000	00000000	65033000	02301705	00000000	00000000	66402400	02400000
66402300	02301672	65402500	02500360	66402600	02600000	00000000	00000000	65402700	02700714	00000000	00000000
65403000	03000525	00000000	00000000	65403100	03100350	00000000	00000000	65404300	04300362	65403200	03200532
00000000	00000000	65403300	03300126	00000000	00000000	65403400	03400000	00000000	00000000	65403500	03500637
00000000	00000000	65035500	03600213	00000000	00000000	66035600	03600207	66403600	03600210	65403700	03700321
00000000	00000000	65404000	04001274	00000000	00000000	65404100	04100042	00000000	00000000	65404200	04200132
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
65404500	04500612	65037600	04401376	65404400	04400000	65404600	04601704	00000000	00000000	65404700	04700291
65405000	05001723	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	66041200	05100023
00000000	00000000	00000000	00000000	66405100	05100022	66024500	00200612	66041600	05200334	00000000	00000000
65405200	05200006	66041300	00201101	66042300	05300032	66405400	05400000	66405500	05500000	64405300	05300003
66041700	00201102	00000000	00000000	65407100	07100000	65406300	06300000	65407300	07300000	64021400	00200323
65406100	06100000	65020200	00200333	65024300	00200363	65024400	00200365	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000

SEGZERO

00000000	00001134	00000000	00000077	00000000	00001137	00000000	00000430	00000000	00001132	00000000	00000044
00000000	00000001	00000000	57500014	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000

NUMBER OF ERRORS DETECTED = 0. COMPILATION TIME = 701 SECONDS.

PRT SIZE = 280; TOTAL SEGMENT SIZE = 15832 WORDS; DISK SIZE = 630 SEGS; NO. PGM. SEGS = 62

ESTIMATED CORE STORAGE REQUIRED = 24408 WORDS.

ESTIMATED AUXILIARY MEMORY REQUIRED = 0 WORDS.

NUMBER OF CARD-IMAGES PROCESSED = 3685.

A
0005 *00000471*
00000539 00000542 00000549 00000555

A
0002 *00000597*
00000599

A
0002 *00000604*
00000606 00000611

A
0002 *00000617*
00000627

ABSX
0004 *00000318*
00000331 00000332 00000334 *00000334* 00000334 00000335 *00000335* 00000335 00000336 00000337

ADDADDR
0010 *00001165*
00001178 *00001228*

ADDR
0037 *00002816*
00002821 00002849 *00002849* 00002854 00002858 00002880 *00002881* 00002881 00002915

ALFA
0002 *00000091*
00000515 00000763 00001556 00001752 00001849 00001854 00001972 00001977 00001995 00001996 00002092
00002112 00002314 00002662 00002666 00002898 00003204 00003326

ALFACONST
0002 *00000196*
00000751 00001004 00001556 00001948 00001964 00001995 00002314 00002662 00002666 00002898 00003204

ALFANUM

0002 *00000860*

00000893 00000899 00000903

ALFANUM1?

0002 *00000860*

00000893 00000899 00000903

ALFATYPE

0002 *00000163*

00000514 00000754 00001508 00001523 00001527 00001556 00002314 00002662 00002666 00002898 00003204

ALGOLCARD

0002 *00000130*

00000362 00000363 00000364 00000365 00000487

ALGOLCNT

0002 *00000132*

00000281 00000284 00000290 00000293 00000306 00000311 00000323 00000326 *00000326* 00000329 00000330
00000330 00000330 00000340 *00000340* 00000341 *00000341* 00000341 00000344 *00000344* 00000345 *00000345*
00000345 00000346 *00000346* 00000346 *00000365* *00000487* *00001809* 00001809 *00001809* 00001809 *00001809*
00001809 00001882 *00001882* 00001882 *00001902* 00001902 *00001902* 00001902 *00001902* 00001902 *00001902*
00001903 00001903 *00001904* 00001904 *00001904* 00001904 00001905 *00001905* 00001905 *00001905* *00001906*
00001906 00001907 *00001907* 00001907 *00001907* 00001908 *00001908* 00001908 *00001908* 00001908 00001909
00001909 00001909 *00001909* 00001909 *00001909* 00001909 *00001909* 00001912 *00001912* 00001912 *00001912*
00001912 *00001912* 00001912 *00001913* 00001913 *00001913* 00001913 *00001913* 00001913 *00001913* 00001914 *00001914*
00001914 00001914 *00001920* 00001920 00001927 *00001927* 00001927 *00001927* *00001928* 00001928 *00001928*
00001928 00001929 *00001929* 00001929 *00001929* 00001931 *00001931* 00001931 *00001940* 00001940 00001950
00001950 00001950 *00001950* 00001950 *00001950* 00001950 *00001951* 00001951 *00001951* 00001951 00001952
00001952 00001952 *00001952* 00001957 *00001959* 00001959 00001960 *00001960* 00001960 00001961 *00001961*
00001961 *00001961* 00001961 *00001961* 00001961 *00001962* 00001962 *00001962* 00001962 00001966 *00001966*
00001966 *00001966* 00001966 *00001966* *00001967* 00001967 *00001967* *00001968* 00001968 *00001968* 00001971
00001971 00001971 *00001975* 00001975 *00001975* 00001975 *00001976* 00001976 *00001976* 00001976 00001977
00001977 00001977 *00001977* 00001977 *00001978* 00001978 *00001978* 00001978 *00001978* 00001979 *00001979*
00001979 *00001984* 00001984 00001990 *00001990* 00001990 00001991 *00001991* 00001991 *00001994* 00001994
00001994 00001994 *00001995* 00001995 *00001995* 00001995 00001996 *00001996* 00001996 *00001996* 00001996
00001997 00001997 *00001997* 00001997 *00001997* 00001997 *00001997* 00001997 00001998 *00001998* 00002001
00002001 00002001 *00002001* 00002001 *00002003* 00002003 00002012 *00002012* 00002012 *00002012* 00002012
00002012 00002012 *00002013* 00002013 *00002013* 00002014 *00002014* 00002014 *00002014* 00002014 00002016
00002016 00002016 *00002026* 00002026 *00002027* 00002027 *00002028* 00002028 00002029 *00002029* *00002030*
00002030 00002032 *00002032* 00002032 *00002036* 00002036 *00002036* 00002036 00002037 *00002037* *00002038*
00002038 *00002038* 00002038 *00002039* 00002039 *00002039* 00002039 *00002039* 00002039 00002048 *00002048*
00002048 00002065 *00002065* 00002065 00002067 *00002067* 00002067 *00002067* 00002067 *00002067* 00002067
00002068 *00002068* 00002068 *00002068* 00002068 *00002068* *00002076* 00002076 *00002080* 00002080 *00002090*
00002090 *00002090* 00002090 *00002090* 00002090 00002101 *00002101* 00002101 *00002101* 00002101 *00002101*
00002101 *00002101* 00002101 *00002108* 00002108 *00002115* 00002115 *00002129* 00002129 00002130 *00002130*

00002130 *00002130* 00002130 *00002130* 00002131* 00002131 *00002131* 00002131 *00002131* 00002131 *00002139*
00002139 *00002146* 00002146 *00002146* 00002146 *00002146* 00002146 *00002146* 00002146 00002153 *00002153*
00002154 *00002154* 00002154 *00002154* 00002154 00002167 *00002167* 00002167 *00002167* 00002167 *00002167*
00002167 *00002167* 00002171 *00002171* 00002171 *00002171* 00002171 *00002171* 00002171 *00002171* 00002171 *00002220* 00002220
00002220 00002220 *00002220* 00002220 *00002220* 00002220 00002224 *00002224* 00002224 *00002224* 00002224
00002224 00002224 00002230 *00002230* 00002230 00002248 *00002248* 00002248 00002252 *00002252* 00002252
00002260 *00002260* 00002260 00002268 *00002268* 00002268 00002269 *00002269* 00002269 *00002269* 00002269
00002269 00002269 *00002269* 00002277 *00002277* 00002280 *00002280* 00002280 00002295 *00002295* 00002295
00002295 00002295 *00002297* 00002297 *00002311* 00002311 *00002319* 00002319 *00002319* 00002319 *00002319*
00002319 *00002319* 00002319 *00002332* 00002332 *00002335* 00002335 00002342 *00002342* 00002345 *00002345*
00002345 *00002354* 00002354 00002355 *00002355* 00002355 *00002355* 00002355 *00002355* 00002355 *00002355*
00002363 *00002363* 00002363 *00002377* 00002377 *00002377* 00002377 00002381 *00002381* 00002381 *00002392*
00002392 00002393 *00002393* 00002393 *00002393* 00002403* 00002403 *00002431* 00002431 *00002433* 00002433
00002433 00002433 *00002433* 00002433 00002434 *00002434* 00002434 *00002445* 00002445 *00002445* 00002445
00002447 *00002447* 00002447 *00002447* 00002447 *00002447* 00002447 *00002447* 00002447 *00002448* 00002448
00002456 00002456 *00002456* 00002456 00002457 *00002457* 00002457 *00002457* 00002457 *00002458* 00002458
00002458 00002458 *00002459* 00002459 *00002459* 00002459 00002461 *00002461* *00002474* 00002474 *00002474*
00002474 *00002545* 00002545 *00002545* 00002545 *00002545* 00002584* 00002584 *00002584* 00002584 *00002584*
00002584 *00002584* 00002584 *00002961* 00002961 00002962 *00002962* *00002963* 00002963 *00002967* 00002967 00002968
00002968 00002968 00002969 *00002969* 00002969 *00002970* 00002970 00002973 *00002973* 00002973 *00002976*
00002976 00002977 *00002977* 00002977 00002978 *00002978* 00002978 *00002978* 00002978 *00002981* 00002981
00002982 00002982 *00002982* 00002982 00002984 *00002984* 00002984 *00002984* 00002984 *00002984* 00002988
00002988 00002988 00002992 *00002992* 00002992 00003013 *00003013* 00003013 *00003014* 00003014 *00003015*
00003015 *00003016* 00003016 *00003016* 00003016 00003017 *00003017* 00003020 *00003020* 00003020 *00003024*
00003024 *00003024* 00003024 00003027 00003028 *00003028* 00003029 *00003029* 00003029 *00003030* 00003030
00003031 00003031 *00003031* 00003031 00003032 00003033 *00003033* 00003033 *00003033* 00003036 *00003036*
00003036 *00003036* 00003036 00003038 *00003038* 00003042 *00003042* 00003042 *00003042* 00003042 00003043
00003043 00003044 *00003044* 00003044 00003045 *00003046* 00003046 *00003046* 00003046 00003047 *00003047*
00003048 *00003048* *00003049* 00003049 *00003049* 00003049 *00003050* 00003050 00003053 *00003053* 00003053
00003053 00003056 *00003056* 00003056 *00003056* *00003058* 00003058 *00003058* 00003058 *00003058* *00003060*
00003060 00003061 *00003061* 00003061 *00003061* 00003061 *00003062* 00003062 00003066 *00003066* 00003066
00003169 00003169 00003173 *00003173* 00003173 *00003178* 00003178 *00003187* 00003187 00003190 *00003190*
00003339 *00003339* 00003340 *00003340* 00003340 *00003340* 00003340 00003344 *00003344* 00003346 *00003346*
00003346 *00003349* 00003349 00003350 *00003350* 00003350 00003351 *00003351* 00003351 00003352 *00003352*
00003354 00003354 00003356 *00003356* *00003362* 00003362 *00003364* 00003364 *00003365* 00003365 *00003369*
00003367 *00003369* 00003369 00003376 *00003376* 00003376 00003391 *00003391* 00003391 00003397 *00003397*
00003397 00003399 *00003399* 00003399 *00003400* 00003400 *00003400* 00003411 *00003411* 00003411 *00003412*
00003412 *00003412* 00003413 *00003413* 00003413 00003418 *00003418* *00003419* 00003419 00003420 *00003420*
00003420 00003424 *00003424* 00003425 *00003425* 00003425 *00003425* 00003425 *00003425* 00003441 *00003441*
00003441

ALGOLNAME

0002 *00003457*

00003459 00003460 *00003460* 00003461 *00003461* 00003463 00003563

ALGOLPNT

0002 *00000131*

00000283 00000292 00000308 00000310 00000325 00000330 00000338 00000341 00000343 00000345 00000346
00000365 00000366 *00000487* 00000488 00001809 00001882 00001902 00001903 00001904 00001905 00001906
00001907 00001908 00001909 00001912 00001913 00001914 00001920 00001927 00001928 00001929 00001931
00001940 00001950 00001951 00001952 00001958 00001960 00001961 00001962 00001966 00001967 00001968
00001971 00001975 00001976 00001977 00001978 00001979 00001984 00001990 00001991 00001994 00001995
00001996 00001997 00001998 00002001 00002003 00002012 00002013 00002014 00002016 00002026 00002027

00002028	00002029	00002030	00002032	00002036	00002037	00002038	00002039	00002048	00002065	00002067
00002068	00002076	00002080	00002090	00002101	00002108	00002115	00002129	00002130	00002131	00002139
00002146	00002153	00002154	00002167	00002171	00002220	00002224	00002230	00002248	00002252	00002260
00002268	00002269	00002277	00002280	00002295	00002297	00002311	00002319	00002332	00002335	00002342
00002345	00002354	00002355	00002363	00002377	00002381	00002392	00002393	00002403	00002431	00002433
00002434	00002445	00002447	00002448	00002456	00002457	00002458	00002459	00002461	00002474	00002545
00002584	00002958	00002961	00002962	00002963	00002967	00002968	00002969	00002970	00002973	00002976
00002977	00002978	00002981	00002982	00002984	00002988	00002992	00003013	00003014	00003015	00003016
00003017	00003020	00003024	00003028	00003029	00003030	00003031	00003033	00003036	00003038	00003042
00003043	00003044	00003046	00003047	00003048	00003049	00003050	00003053	00003056	00003058	00003060
00003061	00003062	00003066	00003169	00003173	00003178	00003187	00003190	00003339	00003340	00003344
00003346	00003349	00003350	00003351	00003352	00003354	00003356	00003362	00003364	00003365	00003367
00003369	00003376	00003391	00003397	00003399	00003400	00003411	00003412	00003413	00003418	00003419
00003420	00003424	00003425	00003441							

ALIST

0003 *00000189*

00001051

ANDSY

0002 *00000198*

00000478 00000920 00001640 00001651

ARRAYS

0002 *00000096*

00001195 00002061 00002124 00002725 00002985

ARRAYSY

0002 *00000208*

00000940 00002714

ARRAYVAR

0038 *00002945*

00002956 *00002959* 00002967 *00002967* 00002976 *00002977* 00002991 *00002992* 00003066

ARROW

0002 *00000203*

00000477 00000478 00001184 00001264 00001265 00002688

ARRSTRUCT

0036 *00002649*

00002740 00002744 *00002744*

ARRTYPE

0002 *00000098*

00001200 00002064 00002127 *00002725* 00002743 *00002744* 00002985

ASSIGN

0026 *00002203*

00002210 00002231 00002236

ASSIGNMENT

0002 *00002200*

00002239 00002562 00002586

ASSIGNSY

0002 *00000203*

00001029 00002210 00002211 00002235 00002236 00002425 00002427 00002428

ASTERISK

0002 *00000197*

00000478 00001631 00001638 00001651

AX

0002 *00000580*

00000591 *00000591* 00000592 *00000592* 00000593

A0

0002 *00000579*

00000606 00000608 00000610 *00000627* 00000628 00000647 00000648 00000650 00000651

A1

0002 *00000579*

00000606 00000609 *00000627* 00000628 00000649

A2

0002 *00000580*

00000627 00000628 00000637 00000640 *00000642* 00000642 00000652 00000653

B

0002 *00000604*

00000606 00000611

BEGINNUM

0027 *00002244*

00002247 *00002247* 00002247 00002260

BEGINSY

0002 *00000204*

00000937 00002250 00002588 00003402

BITPATTERN

0002 *00000214*

00001308 00001443 00001566 00001619 00001647 00001668 00001701 00001708 00001728 00001754 00001769
00001796 00001799 00001804 00001818 00002269 00002355 00002393

BLANK

0002 *00000858*

00000869 00000884 00000887 00000892 00000895 00000901 00000903 00000967 00000968 00000971 00000975
00000982 00000983 00000985 00000986 00001004 00001007 00001013 00001027 00001029 00001031 00001033
00001035 00001037 00001040 00001043 00001056 00001061 00001068 00001069 00001072

BLOCK

0002 *00003149*

00003384 00003442 00003516 00003522

BLOCKTAB

0002 *00000138*

00000591 00002795 00003382

BOOLEXP

0002 *00001814*

00001821 00002269 00002355 00002393

BOOLTYPE

0002 *00000163*

00000518 00000526 00001437 00001443 00001563 00001645 00001699 00001799 00001817 00001973 00002269
00002355 00002393

BRACKETSINWITH

0002 *00000079*

00001174 *00002506*

BUF

0041 *00003468*
00003471 00003472

B0

0002 *00000579*
00000606 00000608 00000610

B1

0002 *00000579*
00000606 00000609

C

0002 *00000863*
00000869 00000872 00000884 *00000887* 00000889 00000891 *00000892* 00000893 00000894 *00000895* 00000899
00000900 *00000901* *00000903* 00000903 *00000903* 00000903 00000964 00000966 *00000967* 00000968 *00000968*
00000968 00000969 *00000971* 00000972 00000974 *00000975* 00000976 *00000977* 00000977 00000980 *00000982*
00000983 *00000983* 00000983 00000984 00000985 *00000985* 00000986 *00000986* 00001002 *00001004* *00001007*
00001007 *00001013* *00001027* 00001027 *00001027* 00001028 *00001029* 00001030 *00001031* 00001032 *00001033*
00001034 *00001035* 00001036 *00001037* 00001038 *00001040* 00001041 *00001043* 00001043 *00001043* 00001044
00001045 00001046 00001047 00001048 00001049 00001051 00001055 00001056 *00001056* *00001061* 00001062
00001065 *00001068* 00001069 *00001069* 00001070 *00001072* 00002289 *00003479*

C

0029 *00002289*
00002327 00002329

CALLGEN

0016 *00001737*
00001764 *00001777* 00001798

CARD

0002 *00000035*
00000269 00000886 00001067 00003477

CARDCNT

0002 *00000034*
00000253 00000275 00000362 00000409 00000591 00000886 *00000886* 00000886 00001067 *00001067* 00001152
00001198 00001314 00001418 00001469 00001480 00001487 00001494 00001609 00001864 00001909 00001914
00001962 00002001 00002039 00002101 00002146 00002222 00002432 00002446 *00003477* 00003477 00003529

CARDLENGTH

0002 *00000160*

00000273 00000398 00000886 00000958 00001054 *00001055* 00001056 *00001056* 00001057 *00001058* 00001059
00001067 *00003476* 00003477

CARDPNT

0002 *00000131*

00000270 00000271 00000871 00000885 00000886 *00000886* 00000887 00000892 00000895 00000901 00000903
00000959 00000959 00000960 00000967 00000968 00000971 00000975 00000982 00000983 00000985 00000986
00001004 00001007 00001013 00001027 00001029 00001031 00001033 00001035 00001037 00001040 00001043
00001056 00001061 00001066 00001067 *00001067* 00001068 00001069 00001072 *00003477* 00003477

CASEHASH

0029 *00002287*

00002326

CASEHASH1?

0029 *00002287*

00002326

CASENUM

0029 *00002289*

00002292 *00002292* 00002292 00002345

CASEPART

0037 *00002819*

00002824 *00002857*

CASESTAT

0002 *00002285*

00002348 00002590

CASESY

0002 *00000205*

00000388 00000933 00002590 00002824

CASETAB

0029 *00002288*

00002327 00002329

CASETYPE

0029 *00002289*

00002297 00002298 *00002299* 00002301 00002304 00002317 *00002317* 00002318

CASETYPE

0037 *00002816*

00002875 00002875 00002876 00002880 00002890 00002901 *00002901* 00002903

CASETYPEID

0037 *00002819*

00002863 *00002875*

CH

0002 *00000128*

00000291	00000489	00000872	00000887	00000892	00000895	00000901	00000903	00000967	00000968	00000971
00000975	00000982	00000983	00000985	00000986	00001004	00001007	00001013	00001018	00001027	00001029
00001031	00001033	00001035	00001037	00001040	00001043	00001056	00001061	00001068	00001069	00001072
00001904	00001905	00001906	00001927	00001928	00001929	00001950	00001951	00001952	00001966	00001967
00001968	00002012	00002013	00002014	00002036	00002037	00002038	00002065	00002129	00002230	00002295
00002319	00002431	00002445	00002456	00002457	00002459	00002474	00002545	00002963	00002970	00002978
00003013	00003020	00003049	00003061	00003178	00003340	00003346	00003350	00003352	00003365	00003399
00003419	00003424									

CHANGELINE

0002 *00001878*

00001925

CHAR

0002 *00000089*

00000523	00000693	00001199	00001303	00001357	00001451	00001466	00001598	00001599	00001606	00001607
00001660	00001723	00001792	00001795	00001899	00001972	00001978	00001997	00002064	00002077	00002127
00002140	00002221	00002225	00002318	00002417	00002435	00002449	00002624	00002662	00002666	00002724
00002765	00002781	00002877	00002903							

CHARCNT

0002 *00000132*

00000273	00000398	00000869	00000872	00000885	*00000886*	00000886	00000887	*00000887*	00000892	*00000892*
00000895	*00000895*	00000895	*00000901*	00000901	*00000903*	00000903	00000958	00000959	*00000967*	00000967
00000968	*00000968*	*00000971*	00000971	*00000975*	00000975	*00000982*	00000982	*00000983*	00000983	00000985
00000985	00000985	*00000986*	00000986	00001004	*00001004*	00001007	*00001007*	00001007	00001008	00001013
00001013	*00001027*	00001027	00001029	*00001029*	00001031	*00001031*	00001033	*00001033*	00001033	00001035

00001035 00001035 00001037 *00001037* *00001040* 00001040 00001043 *00001043* 00001043 *00001043* 00001054
00001056 *00001056* *00001059* 00001061 *00001061* 00001061 00001066 00001067 *00001067* 00001068 *00001068*
00001068 *00001069* 00001069 *00001072* 00001072 *00003477* *00003526* 00003529

CHARCONST

0002 *00000197*

00000743 00001017 00001554 00001556 00002314 00002662 00002666 00002776 00002893 00002898 00003204

CHARPNT

0002 *00000129*

00000292 *00000489* 00000871 00000887 00000892 00000895 00000901 00000903 00000967 00000968 00000971
00000975 00000982 00000983 00000985 00000986 00001004 00001007 00001011 00001013 00001018 00001027
00001029 00001031 00001033 00001035 00001037 00001040 00001043 00001056 00001061 00001068 00001069
00001072 00001904 00001905 00001906 00001927 00001928 00001929 00001950 00001951 00001952 00001966
00001967 00001968 00002012 00002013 00002014 00002036 00002037 00002038 00002065 00002129 00002230
00002295 00002319 00002431 00002445 00002456 00002457 00002459 00002474 00002545 00002963 00002970
00002978 00003013 00003020 00003049 00003061 00003178 00003340 00003846 00003350 00003352 00003365
00003399 00003419 00003424

CHARTYPE

0002 *00000163*

00000522 00000745 00001246 00001423 00001556 00002314 00002662 00002666 00002898 00003204

CHECK

0018 *00001881*

00001901 00001903 00001910

CHECK

0011 *00001284*

00001303 00001304 00001310

CHECKEXPR

0002 *00001146*

00001154 00001198 00001418 00002222 00002432 00002446

CHECKEXPR1?

0002 *00001146*

00001198 00001418 00002222 00002432 00002446

CHECKEXPR2?

0002 *00001146*

00001198 00001418 00002222 00002432 00002446

CHECKOPTION
0002 *00000159*

00001047 00001198 00001303 00001901 00002221 00002432 00002446 *00003478*

CHECKTYPES
0002 *00000662*

00000696 00001199 00001357 00001599 00001607 00001660 00001723 00001792 00001795 00002077 00002140
00002225 00002318 00002435 00002449 00002903

CHECKTYPES1?
0002 *00000662*

00001199 00001357 00001599 00001607 00001660 00001723 00001792 00001795 00002077 00002140 00002225
00002318 00002435 00002449 00002903

CHECKTYPES2?
0002 *00000662*

00001199 00001357 00001599 00001607 00001660 00001723 00001792 00001795 00002077 00002140 00002225
00002318 00002435 00002449 00002903

COLON
0002 *00000203*

00000477 00001028 00001980 00001985 00002336 00002337 00002547 00002549 00002842 00002866 00002911
00002912 00003105 00003258 00003260 00003261 00003314

COMMA
0002 *00000202*

00000479 00001209 00001359 00001360 00001610 00001611 00001850 00001855 00001858 00001861 00001889
00001921 00001947 00002004 00002074 00002078 00002113 00002137 00002175 00002332 00002334 00002521
00002679 00002734 00002839 00002841 00002910 00003104 00003186 00003187 00003188 00003249 00003259
00003504

COMPSTAT
0002 *00002242*

00002262 00002588 00003429

CONCAT
0002 *00001091*

00001499 *00001835* 00001874

CONST
0002 *00000064*

00000526 00000534 00000761 00001301 00001325 00001395 00001397 00001556 00001896 00002088 00002207
00002314 00002412 00002561 00002662 00002666 00002670 00002898 00003092 00003119 00003125 00003204
00003205 00003359

CONSTANT
0002 *00000733*

00000779 00001556 00001621 00002314 00002623 00002627 00002662 00002666 00002898 00003204 00003220

CONSTANT1?
0002 *00000733*

00001556 00002314 00002662 00002666 00002898 00003204

CONSTANT2?
0002 *00000733*

00001556 00002314 00002662 00002666 00002898 00003204

CONSTSY
0002 *00000209*

00000941 00003194 00003406

CONSTTAB
0002 *00000118*

00000536 00000766 00001400 00001556 00002314 00002662 00002666 00002898 00003204 00003211

CONSTTABTOP
0040 *00003157*

00003162 00003438

CONTYPE
0029 *00002289*

00002314 00002314 *00002314* 00002315 00002317 00002318

CONVAL
0029 *00002289*

00002314 *00002314* 00002314 *00002314* 00002314 *00002314* 00002314 00002319 00002323 00002326 00002327
00002329

CTYPE
0037 *00002816*
00002898 00002898 *00002898* 00002899 00002901 00002903

CTYPE
0040 *00003151*
00003204 00003204 *00003204* 00003205 00003233 00003234 00003264 00003265

CURFUNC
0002 *00000179*
00002229 00003383 *00003383**00003387*

CURKIND
0039 *00003072*
*00003089**00003092* 00003099 00003107 00003119 00003125 00003133 00003136

CURLNGTH
0002 *00000864*
00000891 00000893 *00000895* 00000895 00000897 00000899 *00000901* 00000901 00000905 00000906 00000958
00000960 00000961 *00001004* 00001011 *00001012* 00001012 00001015 00001016 00001020 00001022 00001556
00001948 00001955 00001961 00001995 00002314 00002662 00002666 00002898 00003204 00003485 00003496

CURLEVEL
0002 *00000173*
00000443 00001168 00001180 00001321 00001347 00001391 00001429 00001556 00001883 00001892 00001941
00002032 00002055 00002066 00002084 00002118 00002128 00002161 00002229 00002314 00002407 00002414
00002415 00002489 00002557 00002655 00002662 00002666 00002675 00002676 00002696 00002793 00002862
00002870 00002898 00003098 00003100 00003111 00003138 00003168 00003169 00003200 00003204 00003214
00003228 00003235 00003255 00003267 00003272 00003281 00003282 00003294 00003295 00003309 00003312
00003320 00003328 00003340 00003341 00003346 00003350 00003352 00003365 00003368 00003374 *00003380*
00003380 00003381 00003382 00003385 *00003387* 00003387 00003388 00003409 00003419 00003424 00003441
00003512

CURMODE
0002 *00000182*
00001277 00001308 *00001366**00001388**00001401**00001443* 00001564 *00001566**00001587**00001619* 00001630
00001648 00001668 *00001670* 00001686 00001702 00001709 00001728 *00001730* 00001754 00001769 00001796
00001799 00001804 00001818 00002269 00002355 00002393

CURNAME1
0002 *00000863*
*00000891**00000894* 00000894 00000905 00000909 00000910 00000911 00000912 00000913 00000914 00000915

00000916	00000917	00000918	00000919	00000920	00000921	00000922	00000923	00000924	00000925	00000926
00000927	00000928	00000929	00000930	00000931	00000932	00000933	00000934	00000935	00000936	00000937
00000938	00000939	00000940	00000941	00000942	00000943	00000944	00000945	00000946	00000947	00000949
00000951	00001224	00001321	00001347	00001391	00001409	00001414	00001425	00001426	00001428	00001429
00001439	00001445	00001457	00001458	00001460	00001475	00001483	00001490	00001498	00001500	00001505
00001510	00001515	00001520	00001524	00001530	00001531	00001532	00001533	00001534	00001535	00001556
00001883	00001892	00001941	00002032	00002055	00002084	00002118	00002153	00002161	00002314	00002407
00002489	00002557	00002567	00002568	00002570	00002571	00002572	00002573	00002574	00002575	00002576
00002578	00002579	00002581	00002582	00002655	00002662	00002666	00002675	00002696	00002705	00002834
00002862	00002864	00002870	00002898	00003098	00003111	00003200	00003204	00003228	00003255	00003277
00003280	00003281	*00003286*	00003294	00003309	00003320	00003372	00003485	00003493	00003494	00003499

CURNAME2

0002 *00000863*

00000891	00000900	*00000900*	00000947	00000949	00000951	00001224	00001321	00001347	00001391	00001429
00001510	00001520	00001556	00001883	00001892	00001941	00002032	00002055	00002084	00002118	00002161
00002314	00002407	00002489	00002557	00002569	00002577	00002580	00002655	00002662	00002666	00002675
00002696	00002705	00002834	00002862	00002864	00002870	00002898	00003098	00003111	00003200	00003204
00003228	00003255	00003277	*00003280*	00003281	*00003286*	00003294	00003309	00003320	00003372	

CURSY

0002 *00000180*

00000384	00000385	00000388	00000705	00000709	00000722	00000735	00000736	00000739	00000743	00000747
00000751	00000756	*00000908*	*00000909*	*00000915*	*00000930*	*00000937*	*00000943*	*00000947*	*00000949*	*00000951*
00000953	*00000954*	*00000955*	00000957	*00000966*	*00000973*	*00000982*	00000991	*00001004*	*00001017*	*00001027*
00001028	*00001029*	00001030	*00001031*	00001032	*00001033*	00001034	*00001035*	00001056	*00001037*	00001038
00001184	00001188	00001209	00001210	00001212	00001215	00001222	00001265	00001293	00001319	00001345
00001359	00001360	00001361	00001363	00001379	00001383	00001384	00001389	00001411	00001417	00001420
00001421	00001429	00001434	00001435	00001441	00001448	00001452	00001462	00001471	00001472	00001477
00001489	00001492	00001536	00001554	00001556	00001558	00001569	00001574	00001578	00001582	00001585
00001600	00001610	00001611	00001612	00001614	00001631	00001633	00001638	00001639	00001640	00001643
00001651	00001680	00001682	00001691	00001696	00001697	00001704	00001711	00001712	00001748	00001751
00001756	00001757	00001758	00001759	00001760	00001761	00001771	00001772	00001775	00001776	00001781
00001782	00001845	00001850	00001855	00001858	00001861	00001866	00001873	00001883	00001885	00001887
00001889	00001890	00001921	00001922	00001923	00001941	00001943	00001945	00001947	00001948	00001969
00001980	00001985	00002004	00002005	00002006	00002032	00002040	00002041	00002050	00002053	00002074
00002078	00002082	00002098	00002099	00002109	00002113	00002116	00002137	00002143	00002144	00002156
00002159	00002175	00002177	00002178	00002181	00002182	00002210	00002212	00002236	00002250	00002253
00002254	00002255	00002256	00002258	00002270	00002273	00002275	00002279	00002300	00002303	00002307
00002309	00002314	00002332	00002334	00002336	00002337	00002339	00002342	00002343	00002344	00002356
00002359	00002361	00002382	00002383	00002384	00002385	00002388	00002390	00002405	00002425	00002428
00002429	00002437	00002438	00002441	00002443	00002450	00002453	00002470	00002487	00002521	00002522
00002524	00002536	00002547	00002549	00002555	00002588	00002589	00002590	00002591	00002592	00002593
00002594	00002595	00002596	00002625	00002653	00002662	00002666	00002667	00002673	00002679	00002680
00002685	00002688	00002691	00002712	00002714	00002717	00002734	00002735	00002736	00002751	00002754
00002756	00002771	00002774	00002776	00002791	00002803	00002804	00002823	00002824	00002825	00002830
00002839	00002841	00002842	00002853	00002860	00002866	00002868	00002888	00002889	00002892	00002893
00002898	00002910	00002911	00002912	00002913	00002917	00002921	00003083	00003087	00003089	00003090
00003096	00003104	00003105	00003109	00003139	00003140	00003142	00003171	00003176	00003187	00003188
00003189	00003191	00003194	00003198	00003202	00003204	00003217	00003218	00003219	00003222	00003226
00003230	00003238	00003239	00003240	00003243	00003249	00003250	00003259	00003260	00003261	00003269
00003270	00003271	00003289	00003291	00003292	00003314	00003318	00003338	00003390	00003392	00003402

00003404 00003406 00003480 00003483 00003487 00003491 00003504 00003505 00003506 00003507 00003519
00003522

CURTYPE

0002 *00000181*

00000704 00000716 00000717 00001177 *00001177* *00001181* 00001193 00001194 00001199 *00001200* 00001206
00001207 00001208 00001219 *00001229* *00001230* *00001231* 00001236 *00001239* 00001240 *00001246* *00001261*
00001262 00001266 *00001329* 00001337 *00001340* *00001341* *00001342* *00001352* *00001354* *00001355* 00001357
00001382 *00001401* *00001411* 00001411 00001412 00001419 *00001423* *00001429* 00001429 *00001429* *00001437*
00001441 *00001441* 00001442 *00001443* 00001451 *00001455* 00001466 00001467 00001468 *00001473* *00001477*
00001477 00001478 *00001481* *00001485* 00001485 00001488 00001492 *00001492* 00001495 *00001496* *00001503*
00001508 *00001513* *00001518* *00001523* *00001527* 00001536 *00001536* 00001537 *00001538* *00001544* *00001550*
00001551 *00001552* *00001556* 00001556 *00001556* 00001562 *00001563* 00001563 *00001571* *00001587* 00001597
00001598 00001599 00001605 00001606 00001607 *00001618* 00001633 00001645 00001655 00001657 00001659
00001660 *00001661* 00001663 *00001665* *00001666* 00001666 00001688 00001693 00001699 00001718 00001720
00001722 00001723 *00001724* 00001726 *00001726* 00001750 00001787 00001788 00001791 00001792 00001794
00001795 *00001799* 00001817 00001839 00001840 00001848 00001849 00001853 00001854 00001857 00001860
00001863 *00001872* 00001883 *00001883* 00001884 00001898 00001912 00001913 *00001941* 00001941 *00001941*
00001942 00001970 00001973 00001983 00001989 00002032 *00002032* 00002034 00002077 00002091 00002092
00002112 00002140 00002165 00002168 00002209 00002225 00002269 00002271 00002274 00002297 00002355
00002357 00002360 00002393 00002435 00002439 00002442 00002449 00002451 00002454 00002495 00002496
00002502 00002503

CURVAL

0002 *00000862*

00000966 *00000968* 00000968 *00000974* 00000974 *00000994* 00000994 00000997 *00000999* *00001018* *00001023*
00001556 00002314 00002472 00002474 00002539 00002545 00002662 00002666 00002898 00003178 00003179
00003181 00003184 00003204

CVAL

0037 *00002818*

00002898 00002898 *00002898* 00002898 *00002898* 00002898 *00002898* 00002898 00002902 00002906 00002907

CVAL

0040 *00003152*

00003204 00003204 *00003204* 00003204 *00003204* 00003204 *00003204* 00003204 *00003204* 00003206 00003211
00003213

CX

0002 *00000863*

00001043 00001044 00001046 00001047 00001048 00001049 00001050 00001053

C1

0040 *00003153*

00003277 00003286

C2
0040 *00003153*
00003277 00003286

D
0034 *00002483*
00002502 00002503 00002504 00002505 00002506 00002513 00002514

DECL
0002 *00000234*

DECL
0002 *00000586*
00000592

DECLAREVARS
0002 *00002938*
00003067 00003272 00003368

DISPLAY
0002 *00000074*
00000444 00001170 00001321 00001347 00001391 00001429 00001556 00001883 00001892 00001941 00002032
00002055 00002084 00002118 00002161 00002314 00002407 00002489 00002514 00002557 00002655 00002662
00002666 00002696 00002862 00002870 00002898 00003111 00003204 00003320

DIVSY
0002 *00000198*
00000917 00001643 00001663

DOLLAR
0002 *00000858*
00001041

DDSY
0002 *00000206*
00000910 00002356 00002358 00002359 00002450 00002452 00002453 00002522 00002523 00002524

DOT
0002 *00000203*

00000478 00001030 00001184 00001215 00001234 00001265 00003519 00003522

DOUBLEDOT
0002 *00000202*

00000480 00001031 00001600 00002625 00002662 00002666

DOWN
0032 *00002400*

00002438 00002458

DOWNTOSY
0002 *00000207*

00000944 00002438

DUMMY
0013 *00001408*

DUMMY
0020 *00002011*

DUMPOPTION
0002 *00000159*

00000364 *00001048*

EDITION
0002 *00000028*

00000494

ELSESY
0002 *00000205*

00000931 00002253 00002279 00002382

EMPTYSET
0002 *00000164*

00000532 00000671 00001199 00001357 00001587 00001599 00001607 00001660 00001723 00001792 00001795

00002077 00002140 00002225 00002318 00002435 00002449 00002903

ENDOFINPUT

0002 *00000185*

00000269 00000886 00001067 00003477 *00003526*

ENDSY

0002 *00000204*

00000915 00002256 00002257 00002258 00002309 00002339 00002344 00002803 00002804

EDF

0041 *00003467*

00003471 *00003474*

EQLSY

0002 *00000200*

00000479 00000929 00001760 00001771 00001772 00001781 00003202 00003230

EQUAL

0002 *00000858*

00001028 00001032 00001036

ERR

0002 *00000156*

00000397 00003684 00003686

ERRINX

0002 *00000134*

00000268 00000398 *00000398* 00000399 00000400 00000402 *00000402* *00000413* 00000886 00001067 00003477
00003530

ERRLINE

0002 *00000133*

00000400 00000410 00000412 00000486

ERRNUM

0002 *00000394*

00000397 00000401 00000402 00000403

ERRNUM
0036 *00002644*

00002645

ERROR
0002 *00000393*

00000404	00000461	00000508	00000512	00000516	00000520	00000524	00000527	00000529	00000533	00000542
00000544	00000545	00000546	00000555	00000557	00000558	00000563	00000566	00000568	00000693	00000722
00000726	00000744	00000752	00000753	00000771	00000774	00000775	00000776	00000777	00000978	00000988
00000999	00001008	00001015	00001058	00001100	00001172	00001180	00001182	00001191	00001195	00001196
00001198	00001199	00001203	00001204	00001205	00001207	00001208	00001211	00001218	00001220	00001228
00001229	00001230	00001231	00001240	00001245	00001250	00001260	00001262	00001268	00001270	00001271
00001287	00001295	00001298	00001304	00001305	00001309	00001312	00001313	00001314	00001330	00001331
00001332	00001338	00001340	00001341	00001342	00001350	00001351	00001354	00001355	00001357	00001359
00001361	00001362	00001365	00001383	00001385	00001399	00001400	00001411	00001412	00001418	00001419
00001420	00001422	00001429	00001430	00001431	00001434	00001441	00001442	00001447	00001451	00001452
00001454	00001455	00001461	00001465	00001466	00001467	00001468	00001469	00001470	00001471	00001473
00001477	00001478	00001479	00001480	00001485	00001486	00001487	00001488	00001492	00001493	00001494
00001495	00001502	00001507	00001512	00001517	00001522	00001526	00001535	00001536	00001537	00001549
00001551	00001552	00001556	00001560	00001563	00001565	00001571	00001576	00001578	00001579	00001587
00001594	00001598	00001599	00001602	00001606	00001607	00001609	00001610	00001613	00001616	00001621
00001628	00001638	00001639	00001640	00001642	00001643	00001649	00001650	00001651	00001652	00001653
00001660	00001663	00001669	00001679	00001682	00001689	00001696	00001697	00001703	00001704	00001710
00001711	00001712	00001713	00001714	00001716	00001723	00001729	00001742	00001745	00001746	00001755
00001758	00001757	00001758	00001759	00001760	00001761	00001763	00001764	00001770	00001772	00001776
00001777	00001781	00001782	00001783	00001785	00001792	00001794	00001795	00001797	00001798	00001807
00001817	00001840	00001843	00001849	00001852	00001854	00001857	00001860	00001863	00001864	00001865
00001866	00001867	00001868	00001869	00001870	00001871	00001883	00001884	00001916	00001917	00001918
00001919	00001922	00001941	00001942	00001983	00001987	00001989	00002000	00002005	00002032	00002033
00002035	00002040	00002064	00002066	00002069	00002070	00002071	00002072	00002077	00002092	00002093
00002094	00002095	00002096	00002097	00002098	00002100	00002112	00002127	00002128	00002132	00002133
00002134	00002135	00002140	00002141	00002142	00002143	00002145	00002170	00002172	00002173	00002174
00002177	00002180	00002181	00002211	00002217	00002222	00002225	00002229	00002233	00002254	00002257
00002269	00002271	00002274	00002294	00002299	00002301	00002304	00002314	00002318	00002324	00002329
00002331	00002336	00002340	00002344	00002355	00002357	00002360	00002382	00002383	00002387	00002393
00002414	00002415	00002420	00002421	00002422	00002423	00002426	00002432	00002435	00002436	00002439
00002442	00002446	00002449	00002451	00002454	00002473	00002476	00002496	00002498	00002507	00002515
00002516	00002517	00002518	00002523	00002540	00002544	00002548	00002582	00002585	00002586	00002597
00002624	00002625	00002629	00002630	00002645	00002662	00002666	00002670	00002675	00002678	00002680
00002693	00002704	00002717	00002724	00002727	00002729	00002735	00002736	00002739	00002755	00002761
00002762	00002764	00002775	00002781	00002782	00002784	00002793	00002798	00002799	00002803	00002805
00002832	00002834	00002838	00002842	00002845	00002864	00002866	00002877	00002884	00002885	00002886
00002887	00002888	00002898	00002902	00002903	00002904	00002908	00002911	00002917	00002919	00002995
00003009	00003078	00003082	00003098	00003101	00003102	00003107	00003125	00003126	00003127	00003128
00003129	00003133	00003141	00003179	00003181	00003182	00003186	00003189	00003200	00003204	00003209
00003215	00003216	00003217	00003228	00003236	00003237	00003238	00003253	00003255	00003258	00003260
00003268	00003269	00003284	00003304	00003306	00003309	00003316	00003329	00003330	00003331	00003332
00003335	00003337	00003338	00003381	00003390	00003396	00003403	00003434	00003496	00003500	00003502
00003505	00003507	00003508	00003509	00003510	00003521	00003526	00003688			

ERRORMESS1
0044 *00003576*

00003685

ERRORMESS2
0045 *00003639*

00003687

ERRORS
0003 *00000188*

00003683

EXIT
0011 *00001285*

00001298 *00001362*

EXIT
0028 *00002267*

00002275 *00002281*

EXIT
0026 *00002203*

00002212 *00002238*

EXIT
0037 *00002819*

00002853 *00002922*

EXIT
0030 *00002353*

00002361 *00002366*

EXP 0008 *00000879*

00000985 00000986 *00000986* 00000987 *00000987* 00000989

EXPRESSION
0002 *00001090*

00001149	00001198	00001307	00001381	00001411	00001418	00001441	00001450	00001464	00001477	00001485
00001492	00001536	00001577	00001595	00001603	*00001734*	00001811	00001816	00001838	00001847	00001852
00001857	00001860	00001863	00001970	00001982	00001988	00002002	00002077	00002111	00002140	00002222
00002223	00002269	00002296	00002355	00002393	00002432	00002446				

EXPRLEVEL

0002 *00001095*

00001306 00001306 *00001307* 00001307 *00001739* 00001739 00001740 00001801 *00001801* 00001802 00001816
00001820 *00002269**00002355**00002393*

EXTERNALFILE

0002 *00000071*

00000565

EXTFILE

0038 *00002945*

*00002999**00003005* 00003007 00003025

EXTFILETAB

0002 *00000152*

00003009 00003499

F

0018 *00001880*

00001898 00001899 00001901 00001907 00001908

F

0016 *00001736*

00001750 00001752 00001763 00001767 00001779

F

0015 *00001676*

00001688 00001689 *00001693* 00001694 00001706

F

0019 *00001938*

00001970 00001972 00001975 00001976 00001977 00001978 00001987 00001994 00001995 00001996 00001997

F

0014 *00001627*

00001634 00001635 00001642 00001645 00001651

F

0021 *00002023*
00002034 00002035

FACTOR
0002 *00001370*
00001561 00001622 00001629 00001654

FIELDLIST
0002 *00002616*
00002797 *00002810* 00002915 00002923

FILEHANDLING
0002 *00002020*
00002042 00002572 00002573 00002574 00002575 00002577

FILEID
0019 *00001938*
00001941 00001950 00001951 00001952 00001966 00001967 00001968 00002008

FILEID
0018 *00001880*
00001883 00001904 00001905 00001906 00001927 00001928 00001929

FILENAME
0002 *00000699*
00000704 00000719 00000724 00000725 00000726 00001429 *00001429* 00001429 *00001429* 00001430 *00001883*
00001883 *00001883* 00001883 *00001883* 00001883 00001941 *00001941* 00001941 *00002008* 00002012 00002013
00002014 *00002032* 00002032 00002033 00002036 00002037 00002038

FILEPARAM
0002 *00000702*
00000727 00001429 00001883 00001941 00002032

FILEPARAM1?
0002 *00000702*
00001429 00001883 00001941 00002032

FILES
0002 *00000101*
00000717 00001237 00001429 00001883 00001884 00001941 00001942 00002032 00002035 00002739 00002761

00002765 00002845 00002974 00003122

FILESY

0002 *00000208*

00000936 00002751

FILETAB

0002 *00000154*

00002997 00003012 00003416 00003419 00003424

FILETYPE

0002 *00000102*

00001239 *00002766* 00003040 00003041 00003051 00003056 00003058

FINIS

0002 *00000618*

00000619

FINIS

0002 *00000865*

*00001005**00001007**00001008* 00001009 00001014 *00001063**00001070* 00001071

FIRST

0002 *00002941*

00002950

FIRST

0012 *00001373*

00001428 00001431 *00001460* 00001465 *00001591* 00001593 *00001593*

FIRST

0029 *00002290*

*00002308**00002313* 00002313

FIRST

0037 *00002817*

00002827 00002829 *00002829**00002895* 00002897 *00002897*

FIRST

0039 *00003073*

*00003093**00003095* 00003095

FIRST

0036 *00002650*

00002718 00002720 *00002720*

FIRSTADDR

0002 *00002618*

FIRSTADDR

0002 *00002812*

00002821

FIRSTDIM

0038 *00002945*

00002979 00002981 *00002981*

FIRSTFILE

0040 *00003158*

00003166 00003415 00003427

FIRSTLAB

0002 *00000125*

00002538 00003160 *00003161* 00003180 00003433 *00003437*

FIRSTPARAM

0039 *00003072*

00003082 00003145

FIRSTSYM

0011 *00001283*

00001305 00001309

FIRSTSYM

0016 *00001738*

00001743 00001806

FIRSTVAR

0002 *00000143*

00003247 00003266

FIRSTWITHSYM

0002 *00000076*

00001171 *00002505*

FLOATING

0002 *00000090*

00000511 00000771 00001556 00001635 00001642 00001689 00001694 00001763 00001899 00001901 00001908
00001972 00001976 00001987 00001994 00002298 00002314 00002662 00002666 00002898 00003204

FNAME

0038 *00002946*

00003000 00003001 00003005 00003015 00003017 00003030

FNLENGTH

0038 *00002947*

00003001 *00003001* 00003002 00003015 00003017 00003030

FNSTART

0038 *00002947*

00003001 00003015 00003017 00003030

FORM

0002 *00000084*

*00000511**00000515**00000519**00000523* 00000669 00000678 00000684 00000687 00000689 00000717 00000763
00000770 00001195 00001199 00001220 00001237 00001242 00001248 00001303 00001357 00001382 00001411
00001419 00001429 00001441 00001451 00001466 00001477 00001485 00001492 00001536 00001556 00001598
00001599 00001606 00001607 00001634 00001659 00001660 00001688 00001693 00001722 00001723 00001750
00001791 00001792 00001794 00001795 00001840 00001849 00001854 00001857 00001880 00001883 00001883
00001884 00001898 00001941 00001942 00001970 00001983 00001989 00002032 00002034 00002061 00002064
00002077 00002092 00002112 00002124 00002127 00002140 00002165 00002221 00002225 00002298 00002314
00002318 00002417 00002435 00002449 00002496 00002624 00002631 00002662 00002666 00002724 00002739
00002761 00002765 00002781 00002845 00002877 00002898 00002903 00002974 00002985 00002993 00003039
00003122 00003204 00003326

FORMAL

0002 *00000069*

00000761 00001289 00001325 00001395 00001556 00001896 00002088 00002207 00002314 00002412 00002561

00002662 00002666 00002898 *00003136* 00003204

FORMALPROC

0011 *00001284*

00001289 00001291 00001365

FORMERFIRSTLAB

0040 *00003158*

00003160 00003437

FORSTAT

0002 *00002397*

00002463 00002593

FORSY

0002 *00000208*

00000916 00002593

FORWARDDEF

0002 *00000070*

00003299 00003301 00003374

FOUND

0002 *00000421*

00000433 00000435 00000446 00000712 00000759 *00001224* 00001224 00001225 00001321 *00001321* 00001322
00001347 *00001347* 00001347 00001348 00001391 *00001391* 00001391 00001392 *00001429* 00001429 00001556
00001556 00001556 *00001883* 00001883 00001892 *00001892* 00001893 *00001941* 00001941 *00002032* 00002032
00002055 *00002055* 00002055 00002056 00002084 *00002084* 00002085 *00002118* 00002118 00002119 00002161
00002161 00002161 00002162 00002204 00002314 *00002314* 00002314 *00002407* 00002407 00002408 00002489
00002489 00002490 00002557 *00002557* 00002557 00002558 00002655 *00002655* 00002655 00002656 00002662
00002662 00002662 *00002662* *00002666* 00002666 *00002666* 00002666 *00002696* 00002696 00002697 00002862
00002862 00002863 00002870 *00002870* 00002871 *00002898* 00002898 *00003111* 00003111 00003112 00003204
00003204 00003204 *00003281* 00003281 00003283 00003294 *00003294* 00003296 00003320 *00003320* 00003321

FUN

0040 *00003154*

00003291 00003303 00003304 00003311 00003316 00003334 00003339 00003383

FUNC

0002 *00000065*

00000548 00001352 00001404 00002562 00002954 00002961 00003090 00003304 00003311

FUNCSY

0002 *00000210*

00000949 00003087 00003090 00003289 00003291

F1

0002 *00000660*

00000669 00000671 00000672 00000674 00000678 00000680 00000684 00000686 00000687 00000692 00000693
 00001199 *00001199* 00001199 *00001199* 00001199 *00001199* *00001357* 00001357 *00001357* 00001357 *00001357*
 00001599 *00001599* 00001599 *00001599* 00001599 *00001599* *00001607* 00001607 *00001607* 00001607 *00001607*
 00001607 *00001607* 00001607 00001660 *00001660* 00001660 *00001660* 00001660 *00001660* *00001723* 00001723
 00001723 00001723 *00001723* 00001723 *00001723* 00001723 *00001792* 00001792 *00001792* 00001792 *00001792*
 00001795 *00001795* 00001795 *00001795* 00001795 00002077 *00002077* 00002077 *00002077* 00002077 *00002077*
 00002140 00002140 *00002140* 00002140 *00002140* 00002140 00002225 *00002225* 00002225 *00002225* 00002225
 00002225 00002225 00002318 *00002318* 00002318 *00002318* 00002318 *00002318* 00002318 *00002318* *00002435*
 00002435 *00002435* 00002435 *00002435* 00002435 *00002435* 00002449 *00002449* 00002449 *00002449* 00002449
 00002449 00002449 00002903 *00002903* 00002903 *00002903* 00002903 *00002903* 00002903 *00002903* 00002903

F2

0002 *00000660*

00000669 00000670 00000671 00000672 00000674 00000678 00000680 00000684 00000688 00000689 00000692
 00000693 *00001199* 00001199 *00001199* 00001199 *00001199* 00001199 00001357 *00001357* 00001357 00001599
 00001599 00001599 *00001599* 00001599 00001607 *00001607* 00001607 *00001607* 00001607 *00001607* 00001607
 00001660 *00001660* 00001660 *00001660* 00001660 *00001660* 00001660 *00001723* 00001723 *00001723* 00001723
 00001723 00001723 00001792 *00001792* 00001792 *00001792* 00001792 *00001792* 00001792 *00001792* 00001792
 00001795 *00001795* 00001795 *00001795* 00001795 *00001795* 00002077 *00002077* 00002077 *00002077* 00002077
 00002077 00002140 *00002140* 00002140 *00002140* 00002140 *00002140* 00002225 *00002225* 00002225 *00002225* 00002318*
 00002318 *00002318* 00002318 *00002318* *00002435* 00002435 *00002435* 00002435 00002449 *00002449* 00002449
 00002449 00002449 *00002903* 00002903 *00002903* 00002903 *00002903* 00002903 *00002903* 00002903

GEN

0002 *00000279*

00000301 00000330 00000341 00000345 00000346 00001135 00001137 00001809 00001882 00001902 00001903
 00001904 00001905 00001906 00001907 00001908 00001909 00001912 00001913 00001914 00001920 00001927
 00001928 00001929 00001931 00001940 00001950 00001951 00001952 00001960 00001961 00001962 00001966
 00001967 00001968 00001971 00001975 00001976 00001977 00001978 00001979 00001984 00001990 00001991
 00001994 00001995 00001996 00001997 00001998 00002001 00002003 00002012 00002013 00002014 00002016
 00002026 00002027 00002028 00002029 00002030 00002032 00002036 00002037 00002038 00002039 00002048
 00002067 00002068 00002076 00002080 00002090 00002101 00002108 00002115 00002130 00002131 00002139
 00002146 00002153 00002154 00002167 00002171 00002220 00002224 00002248 00002252 00002260 00002268
 00002269 00002277 00002280 00002295 00002297 00002311 00002319 00002332 00002335 00002342 00002345
 00002354 00002355 00002363 00002377 00002381 00002392 00002393 00002403 00002431 00002433 00002434
 00002445 00002447 00002448 00002456 00002457 00002458 00002459 00002461 00002474 00002545 00002584
 00002958 00002961 00002962 00002963 00002967 00002968 00002969 00002973 00002976 00002977 00002978
 00002981 00002982 00002984 00002988 00002992 00003013 00003014 00003015 00003016 00003017 00003020
 00003024 00003028 00003029 00003030 00003031 00003033 00003036 00003038 00003042 00003043 00003044
 00003046 00003047 00003048 00003049 00003050 00003053 00003056 00003058 00003060 00003061 00003062
 00003066 00003169 00003173 00003187 00003190 00003339 00003340 00003344 00003349 00003351 00003354

00003356 00003362 00003364 00003367 00003369 00003376 00003391 00003397 00003400 00003411 00003412
00003413 00003418 00003420 00003425 00003441

GENID
0002 *00000288*

00001904 00001905 00001906 00001927 00001928 00001929 00001950 00001951 00001952 00001966 00001967
00001968 00002012 00002013 00002014 00002036 00002037 00002038 00002065 00002129 00002230 00002295
00002319 00002431 00002445 00002456 00002457 00002459 00002474 00002545 00002963 00002970 00002978
00003013 00003020 00003049 00003061 00003178 00003340 00003346 00003350 00003352 00003365 00003399
00003419 00003424

GENID1?
0002 *00000288*

00001904 00001905 00001906 00001927 00001928 00001929 00001950 00001951 00001952 00001966 00001967
00001968 00002012 00002013 00002014 00002036 00002037 00002038 00002065 00002129 00002230 00002295
00002319 00002431 00002445 00002456 00002457 00002459 00002474 00002545 00002963 00002970 00002978
00003013 00003020 00003049 00003061 00003178 00003340 00003346 00003350 00003352 00003365 00003399
00003419 00003424

GENID2?
0002 *00000288*

00001904 00001905 00001906 00001927 00001928 00001929 00001950 00001951 00001952 00001966 00001967
00001968 00002012 00002013 00002014 00002036 00002037 00002038 00002065 00002129 00002230 00002295
00002319 00002431 00002445 00002456 00002457 00002459 00002474 00002545 00002963 00002970 00002978
00003013 00003020 00003049 00003061 00003178 00003340 00003346 00003350 00003352 00003365 00003399
00003419 00003424

GENID3?
0002 *00000288*

00001904 00001905 00001906 00001927 00001928 00001929 00001950 00001951 00001952 00001966 00001967
00001968 00002012 00002013 00002014 00002036 00002037 00002038 00002065 00002129 00002230 00002295
00002319 00002431 00002445 00002456 00002457 00002459 00002474 00002545 00002963 00002970 00002978
00003013 00003020 00003049 00003061 00003178 00003340 00003346 00003350 00003352 00003365 00003399
00003419 00003424

GENINT
0002 *00000299*

00000312 00001140 00001809 00001902 00001907 00001908 00001909 00001912 00001913 00001914 00001961
00001962 00001975 00001976 00001977 00001978 00001994 00001995 00001996 00001997 00002001 00002039
00002067 00002068 00002090 00002101 00002130 00002131 00002146 00002167 00002171 00002220 00002224
00002269 00002319 00002355 00002393 00002433 00002447 00002584 00002982 00002984 00003042 00003043
00003044 00003056 00003058

GENINT1?
0002 *00000299*

00001809 00001902 00001907 00001908 00001909 00001912 00001913 00001914 00001961 00001962 00001975

GEN3?

0002 *00000279*

00000330	00000341	00000345	00000346	00001809	00001882	00001902	00001903	00001904	00001905	00001906
00001907	00001908	00001909	00001912	00001913	00001914	00001920	00001927	00001928	00001929	00001931
00001940	00001950	00001951	00001952	00001960	00001961	00001962	00001966	00001967	00001968	00001971
00001975	00001976	00001977	00001978	00001979	00001984	00001990	00001991	00001994	00001995	00001996
00001997	00001998	00002001	00002003	00002012	00002013	00002014	00002016	00002026	00002027	00002028
00002029	00002030	00002032	00002036	00002037	00002038	00002039	00002048	00002067	00002068	00002076
00002080	00002090	00002101	00002108	00002115	00002130	00002131	00002139	00002146	00002153	00002154
00002167	00002171	00002220	00002224	00002248	00002252	00002260	00002268	00002269	00002277	00002280
00002295	00002297	00002311	00002319	00002332	00002335	00002342	00002345	00002354	00002355	00002363
00002377	00002381	00002392	00002393	00002403	00002431	00002433	00002434	00002445	00002447	00002448
00002456	00002457	00002458	00002459	00002461	00002474	00002545	00002584	00002958	00002961	00002962
00002963	00002967	00002968	00002969	00002973	00002976	00002977	00002978	00002981	00002982	00002984
00002988	00002992	00003013	00003014	00003015	00003016	00003017	00003020	00003024	00003028	00003029
00003030	00003031	00003033	00003036	00003038	00003042	00003043	00003044	00003046	00003047	00003048
00003049	00003050	00003053	00003056	00003058	00003060	00003061	00003062	00003066	00003169	00003173
00003187	00003190	00003339	00003340	00003344	00003349	00003351	00003354	00003356	00003362	00003364
00003367	00003369	00003376	00003391	00003397	00003400	00003411	00003412	00003413	00003418	00003420
00003425	00003441									

GEQSY

0002 *00000200*

00000477 00000926 00001037 00001758 00001776

GOTSTAT

0002 *00002466*

00002477 00002595

GOTOSY

0002 *00000207*

00000934 00002595

GTRSY

0002 *00000200*

00000477 00000927 00001036 00001759

HASH

0002 *00000417*

00000425	00000453	00000508	00000512	00000516	00000520	00000524	00000527	00000529	00000533	00000542
00000544	00000545	00000546	00000555	00000557	00000558	00000563	00000566	00000568	00001224	00001321
00001347	00001391	00001429	00001556	00001883	00001892	00001941	00002032	00002055	00002084	00002118
00002161	00002314	00002407	00002489	00002557	00002655	00002662	00002666	00002675	00002696	00002834
00002862	00002864	00002870	00002898	00003098	00003111	00003200	00003204	00003228	00003255	00003281

00003294 00003309 00003320

HASH1?

0002 *00000417*

00000508	00000512	00000516	00000520	00000524	00000527	00000529	00000533	00000542	00000544	00000545
00000546	00000555	00000557	00000558	00000563	00000566	00000568	00001224	00001321	00001347	00001391
00001429	00001556	00001883	00001892	00001941	00002032	00002055	00002084	00002118	00002161	00002314
00002407	00002489	00002557	00002655	00002662	00002666	00002675	00002696	00002834	00002862	00002864
00002870	00002898	00003098	00003111	00003200	00003204	00003228	00003255	00003281	00003294	00003309
00003320										

HEADING

0002 *00000241*

00000248	00000254	00000409	00000501	00000633	00000644	00000886	00001044	00001067	00003477	00003529
00003692										

HEADTEXT

0002 *00000133*

00000244	00000246	00000409	00000493	00000494	00000496	00000499	00000501	00000633	00000644	00000886
00001044	00001067	00003477	00003529	00003692						

I

0019 *00001938*

00001954 00001955

I

0033 *00002468*

00002471 00002472 *00002472* 00002472 00002473

I

0035 *00002533*

00002538 00002539 00002540 00002541

I

0037 *00002816*

00002847 00002850 *00002906* 00002907 *00002907* 00002908

I

0038 *00002944*

00002950 00002952