SETL NEWSLETTER NR, 66 FEB 10,1972

REVISION NR, 4, AUGUST 14, 1972. E, MILGROM

```
BBBBB     AAAAAA    L        M     M    SSSSS    EEEEEE   TTTTTTT   L
B    B    A    A    L        MM   MM   S        E           T      L
B    B    A    A    L        M MM M    S        E           T      L
BBBBB     AAAAAA    L        M     M     SSS     EEEEEE      T      L
B    B    A    A    L        M     M        S    E           T      L
B    B    A    A    L        M     M        S    E           T      L
B    B    A    A    L        M     M        S    E           T      L
BBBBB     A    A    LLLLLL   M     M    SSSSS    EEEEEE      T      LLLLLL
```

BALMSETL USERS MANUAL
VERSION 1.0 - FEB.1972

SCOPE OF REVISION NR, 1

THE NEW VERSION OF BALMSETL INCORPORATES CHANGES TO THE FOLLOWING PROCEDURES: APPEND LESSFOK CRASH
4 NEW NAMES HAVE BEEN ADDED TO THE LIST OF KEYWORDS:
IDFROMC LEVEL PROCTRAC STKTRACE

THE PROCEDURE CRASH (INVOKED WHEN A TERMINAL ERROR IS DETECTED) WILL NOW PRINT A BACKWARD TRACE INDICATING THE STRUCTURE OF THE PROCEDURE NESTING AT THE TIME OF ERROR-DETECTION, THE TRACE CONSISTS OF A LIST OF THE NAMES OF THE PROCEDURES, EACH NAME BEING FOLLOWED BY THE VALUES OF THE ARGUMENTS.
THE DEPTH OF THE TRACE = INITIALLY 10 - CAN BE SET BY CHANGING THE VALUE OF THE VARIABLE LEVEL (SEE SECTION 2.2).

SCOPE OF REVISION NR. 2

THE NEW VERSION OF BALMSETL INCORPORATES CHANGES TO THE FOLLOWING PROCEDURES:

BOFSET, SETFUN, SOFN, NELT, DIMINISP

THE FOLLOWING PROCEDURES HAVE BEEN ADDED TO THE SYSTEM:

ERROR, LESFN, DIMFN, LESFNOK, SETFUNN, SETFUNNS, SETFUNS, HEADSIN, TAILSIN, FROMSET, ERRFOR, TAILN, TAILSPEC

THE LANGUAGE INCLUDES THE FOLLOWING ADDITIONAL FEATURES:

- A PROCEDURE DIMFN AND AN OPERATOR LESFN SIMILAR TO DIMF AND LESF BUT FOR MULTIPLE ARGUMENTS

- THE SINISTER CALLS

```
LET F SOF O BE S
LET F OFN LIST(O1,O2,...,ON) BE O
LET F SOFN LIST(O1,O2,...,ON) BE S
LET HEAD T BE O
LET TAIL T BE O
```

- THE ITERATION STATEMENTS

```
FORALL I INR(J,K) SUCH B REPEAT ST
FORALL I INR(J,K) REPEAT ST
```

- THE STATEMENT

```
O FROM S
```

- THE FUNCTION TAILN(T,N) WHICH RETURNS THE TAIL OF A TUPLE T FROM THE N-TH ELEMENT ONWARDS.

- THE PROCEDURE PRTMAP(S) WHICH PRINTS A SET S OF TUPLES AS A MAP.

SCOPE OF REVISION NR. 3

THE NEW VERSION OF BALMSETL INCORPORATES MAJOR CHANGES TO A LARGE NUMBER OF ROUTINES, MOSTLY TO PROVIDE A BETTER INTERFACE WITH SETLB, THE MOST IMPORTANT CHANGES PERTAIN TO INPUT/OUTPUT STATEMENTS (SEE 2.1.10).

THE FOLLOWING PROCEDURES HAVE BEEN ADDED: RANDOM, STRINGOF, PAIRTUP,

THE OPERATOR TYPE HAS BEEN ADDED, TOGETHER WITH THE TYPE-VALUES INTZZ, SETZZ, TUPLZZ, BLANKZZ, STRZZ, LABZZ, SURZZ,

THE VARIABLE CRASHMAX (INITIALLY 5) SPECIFIES THE NUMBER OF ALLOWED SYSTEM CRASHES,

THE SYSTEM HAS GROWN TO A SIZE WHERE IT REQUIRES 65 K(OCTAL) OF CORE INTERCOM USERS BEWARE,,,,,,,,,,,,,

SCOPE OF REVISION NR. 4

THIS VERSION OF THE BALMSETL SYSTEM USES A NEW SWITCHABLE FEATURE OF BALM4 WHICH ENABLES ONE TO USE AN ARBITRARY CHARACTER AS TERMINATOR FOR IDENTIFIERS. ALL THE NAMES OF INTERNAL BALMSETL VARIABLES WHICH ARE NOT USED AT THE USER LEVEL ARE NOW TERMINATED WITH A PERIOD. AT THE END OF THE PROGRAM, THE ABILITY TO TERMINATE NAMES WITH A PERIOD IS REMOVED: ALL THE DOTTED NAMES ARE THEREFORE PROTECTED AGAINST INADVERTENT USE BY A CARELESS USER. THIS METHOD REPLACES TH OLD NAME-PROTECTION SCHEME WHICH MADE USE OF THE FUNCTIONS PROTECT AND DUPL.

THE NON-DOTTED NAMES ARE USED TO IDENTIFY
OPERATORS
PROCEDURES
AND OTHER GLOBAL VARIABLES,

THE COMPILER WILL PREVENT AUTOMATICALLY ONLY THE ATTEMPTED ASSIGNMENT OF A VALUE TO AN OPERATOR, THE USER MUST BE CAREFUL TO AVOID USING NAMES OF PROCEDURES OR OTHER VARIABLES FOR HIS OWN PURPOSES.

CONTENTS

PLEASE REPORT ALL BUGS TO ELIE MILGROM, ROOM 430, PHONE 460-7168

THANK YOU...

0. INTRODUCTION

THE BALMSETL SYSTEM IS AN EXTENSION TO THE BALM4 SYSTEM WHICH INCLUDES MOST OF THE SEMANTICS OF SETL, ALTHOUGH IN A DIFFERENT SYNTACTIC FRAME. THE SYSTEM MAY BE USED BY ITSELF OR AS THE REAR END OF THE SETLB PREPROCESSOR WHICH TRANSLATES THE SYNTAX OF SETLB INTO THAT OF BALMSETL.

SOME ASPECTS OF BALMSETL CORRESPOND VERY CLOSELY TO RESTRCTIONS DUE TO BALM. THE READER IS URGED TO READ SECTION 3 OF THIS MANUAL AND SETL NEWSLETTER NR. 70 WITH THE GREATEST ATTENTION. SOME OF THE SEMANTIC DIFFERENCES BETWEEN BALMSETL AND PURE SETL CAN CAUSE INSIDIOUS BUGS; PARTICULAR ATTENTION SHOULD BE DEVOTED TO THE NAME SCOPING RULES, THE RESTRICTIONS ON LABELS AND GOTOS, THE COPY PROBLEM AND BEGIN- AND DO-BLOCKS.

1. DATA TYPES

THE FOLLOWING SETL TYPES ARE PROVIDED IN BALMSETL:

	PREDICATE:
INTEGER	INTQ
BLANK ATOM	BLANKQ
CHARACTER STRING	STRQ
SET	SETQQ
TUPLE	TUPQ
LABEL	LBLQ

THE FOLLOWING BALM TYPES ARE AVAILABLE IN BALMSETL:

	PREDICATE:
LIST	PAIRQ
VECTOR	VECTQ
IDENTIFIER	IDQ
PROCEDURE	CODEQ
LOGICAL (TRUE,FALSE)	LOGQ

THE FOLLOWING SETL TYPES ARE NOT AVAILABLE IN BALMSETL:

REAL
BIT STRING
SUBROUTINE
FUNCTION

CONSTANTS

THE FOLLOWING CONSTANTS ARE AVAILABLE IN BALMSETL:

NULC	THE EMPTY CHARACTER STRING
NL	THE NULL SET
NULT	THE NULL TUPLE
NILVECT	[NIL] (SEE FUNCTION NEXTELT(S,P))
TRUE	
FALSE,NIL	ALWAYS PRINTED AS NIL

2. FUNCTIONS AND OPERATORS

2.1. CORRESPONDENCE BETWEEN SETL CONSTRUCTS AND BALMSETL CONSTRUCTS

IN WHAT FOLLOWS, WE ADHERE TO THE CONVENTIONS:

O,O1,O2	STAND FOR ANY BALMSETL OBJECTS
A,A1,A2	STAND FOR ANY ATOMS
I,J,K,L	STAND FOR ANY INTEGERS
B,B1,B2	STAND FOR ANY BOOLEANS
C,C1,C2	STAND FOR ANY CHARACTER STRINGS
S,S1,S2	STAND FOR ANY SETS
F,F1,F2	STAND FOR ANY FUNCTIONS
T,T1,T2	STAND FOR ANY TUPLES
ST	STANDS FOR ANY STATEMENT

2.1.1. GENERAL

SETL	BALMSETL
ATOM O1	ATOM O1
NEWAT	NEWAT()
O1 = O2	O1 = O2
O1 EQ O2	O1 EQUAL O2
O1 NE O2	O1 NEQUAL O2
THE UNDEFINED VALUE (OMEGA)	UNDEF
O1 EQ OMEGA	UNDFD O1 , OMEGAP(C1)
O1 NE OMEGA	DFD O1 , NOMEGAP(C1)
TYPE O1 EQ INT	TYPE O1 EQ INTZZ, INTQ(O1)
TYPE O1 EQ STR	TYPE O1 EQ STRZZ, STRQ(O1)
TYPE O1 EQ SET	TYPE O1 EQ SETZZ, SETQQ(O1)
TYPE O1 EQ TUPL	TYPE O1 EQ TUPLZZ, TUPQ(O1)
TYPE O1 EQ BLANK	TYPE O1 EQ BLANKZZ, BLANKQ(C1)
TYPE O1 EQ LAB	TYPE O1 EQ LABZZ, LBLQ(O1)
TYPE O1 EQ SUBR	TYPE O1 EQ SUBRZZ, CODEQ(O1)

2.1.2. INTEGER ARITHMETIC

SETL	BALMSETL
I + J	I + J
I - J	I - J
I * J	I * J
I / J	I / J
I // J	I MOD J, I REM J, I // J
I EXP J	I ↑ J
I LT J	I LT J
I LE J	I LE J
I GT J	I GT J
I GE J	I GE J
I MAX J	I MAX J

I MIN J	I MIN J
ABS I	ABS I

2.1.3. BOOLEANS

SETL	BALMSETL
B1 AND B2	B1 AND B2
B1 OR B2	B1 OR B2
NOT B1	NOT B1 , ¬B1

2.1.4. CHARACTER STRINGS

	SETL	BALMSETL
	DEC C	DEC C
	OCT C	OCT C
	DEC I	DEC I
	OCT I	OCT I
CATENATION	C1 + C2	C1 + C2
REPETITION	I * C	I * C
SUBSTRING	C(I:J)	SUB(C,I,J)
NUMBER OF ELEMENTS OF C		NELT C , # C

2.1.5. SETS

	SETL	BALMSETL
SET FORMER	{O1,O2,...,ON}	GENSET(O1,O2,...,ON)
OTHER FORMERS		SETOF O1 WHERE O2 EL S
		SETOF O1 WHERE O2 EL S SUCH B
		SETOF O1 WHERE I IN(J,K)
		SETOF O1 WHERE I IN(J,K) SUCH B
		O EL S, O ∈ S
	MEMBERSHIP TEST	S1 INCS S2
	S1 INCLUDES S2	ARB S
	ARBITRARY ELEMENT	NELT S , # S
	NUMBER OF ELEMENTS	S WITH O
	S WITH O	S LESS O
	S LESS O	AUGMENT(S,O)
	S = S WITH O	DIMINISH(S,O)
	S = S LESS O	S LESF O
	S LESF O	DIMF(S,O)
		S LESFN LIST(O1,O2,...,ON)
		DIMFN(S,LIST(O1,O2,...,ON))
	POW(S)	POW(S)
	NPOW(S,I)	NPOW(S,I)
	O FROM S	O FROM S
UNION	S1 + S2	S1 + S2 , UNION(S1,S2)

```
INTERSECTION S1 * S2                                  S1 * S2 , INTERCT(S1,S2)
DIFFERENCE   S1 - S2                                  S1 - S2 , SETMNS(S1,S2)
SYM.DIFF     S1 // S2                  S1 // S2, S1 DSLSH S2, SYMDIF(S1,S2)
```

NOTE : IN THE LAST 4 CASES ABOVE, THE USE OF THE SECOND BALMSETL CONSTRUCT IS ADVISED ONLY WHEN ONE IS CERTAIN THAT THE OPERANDS ARE SETS, SINCE NO TYPE-CHECKING IS DONE.....

2.1.6. TUPLES

SETL	BALMSETL
TUPLE FORMER : <O1,O2,...,ON>	GENTUP(O1,O2,...,ON)
	NOTE: CONSTRUCT IRREGULAR TUPLES USING UNDEF
HD T	HEAD T (NOT SINISTER)
TL T	TAIL T (NOT SINISTER)
NUMBER OF ELEMENTS OF T	NELT T , # T
T(I)	T COMP I (NOT SINISTER)
CATENATION T1 + T2	T1 + T2
TAIL OF T FROM N-TH COMPONENT	T TAILN N

2.1.7. FUNCTIONAL APPLICATION

SETL	BALMSETL
F(O)	F OF O
FO	F SOF O
F[O]	F BOF O
NOTE: O MUST BE A SINGLE ARGUMENT	
F(O1,O2,...,ON)	F OFN LIST(O1,O2,...,ON)
F$O1,O2,...,ON$	F SOFN LIST(O1,O2,...,ON)
F[O1,O2,...,ON]	F BOFN LIST(O1,O2,...,ON)

2.1.8. ITERATION STATEMENTS

```
FORALL O EL S SUCH B REPEAT ST
FORALL O EL S REPEAT ST
FORALL O EL T SUCH B REPEAT ST
FORALL O EL T REPEAT ST
FORALL I IN(J,K) SUCH B REPEAT ST
FORALL I IN(J,K) REPEAT ST
FORALL I INR(J,K) SUCH B REPEAT ST
FORALL I INR(J,K) REPEAT ST
       (ITERATION FROM K TO J IN STEPS OF -1)

FOR I=(J,K,L) REPEAT ST
FOR I=(J,K) REPEAT ST
WHILE B REPEAT ST
```

IMPORTANT NOTES:

1. IT IS FORBIDDEN TO MODIFY THE VALUE OF THE ITERATION VARIABLE OR OF THE ITERATION RANGE WITHIN THE BODY OF A FORALL ITERATION.
   THE FOLLOWING IS THUS ILLEGAL:

```
FORALL X EL S REPEAT DO
   .....
   S=S LESS X
   .....
END
```

2. IT IS FORBIDDEN TO TRANSFER OUT OF THE BODY OF A FORALL ITERATION BY MEANS OF A GOTO OR A RETURN STATEMENT, THIS RESTRICTION MAY BE LIFTED IN A LATER VERSION.

2,1,9, QUANTIFIED BOOLEAN EXPRESSIONS

```
EXISTS O EL S SUCH B
EXISTS I IN(K,L) SUCH B
```

NOTE: IF THE VALUE OF SUCH AN EXPRESSION IS TRUE, THEN THE VARIABLE O OR I IS SET EQUAL TO THE FIRST ELEMENT IN THE RANGE FOR WHICH B IS TRUE

```
FORALL O EL S QUANT B
FORALL I IN(K,L) QUANT B
```

NOTE: IF THE VALUE OF SUCH AN EXPRESSION IS FALSE, THEN THE VARIABLE O OR I IS SET EQUAL TO THE FIRST ELEMENT IN THE RANGE FOR WHICH B IS FALSE

2,1,10 INPUT-OUTPUT

ALL INPUT/OUTPUT IS NOW FILE-ORIENTED.

THE FOLLOWING 4 FILES ARE AVAILABLE FOR I/O:
INPUT, INFILE, OUTPUT, OUTFILE

- INPUT IS THE SYSTEM INPUT FILE FROM WHICH THE SOURCE TEXT IS READ BY THE COMPILER.
- INFILE IS THE FILE FROM WHICH DATA CAN BE READ BY MEANS OF A READZZZ STATEMENT, THIS FILE IS EQUAL TO THE FILE INPUT BY DEFAULT.
- OUTPUT IS THE SYSTEM OUTPUT FILE ON WHICH THE OUTPUT FROM THE COMPILER IS WRITTEN, DATA CAN BE WRITTEN ON THAT FILE BY MEANS OF THE PRINT, PRT AND PRTMAP STATEMENTS.
- OUTFILE IS THE FILE ON WHICH DATA CAN BE WRITTEN BY MEANS OF THE WRITSETL STATEMENT, THIS FILE IS EQUAL TO THE FILE OUTPUT BY DEFAULT.

THE FORMAT OF THE I/O STATEMENTS IS AS FOLLOWS:

PRINT(O1,O2,...ON)	PRINTS SETL OBJECTS O1,..ON IN SETL EXTERNAL FORM(SEE BELOW) ON FILE OUTPUT, THE ITEMS ARE SEPARATED BY ONE BLANK.
PRT(O1,O2,...ON)	PRINTS OBJECTS O1,..ON IN BALM EXTERNAL FORM ON FILE OUTPUT.
WRITSETL(O1,O2,...ON)	SIMILAR TO PRINT, BUT DATA IS WRITTEN ON FILE OUTFILE.
PRTMAP(S)	PRINTS ONE SET OF TUPLES AS A MAP ON FILE OUTPUT.
READZZZ(O1,O2,...ON)	READS THE SETL EXTERNAL FORM OR THE BALM EXTERNAL FORM OF N OBJECTS FROM FILE INFILE AND STORES THE CORRESPONDING VALUES IN THE VARIABLES O1,..ON ITEM DELIMITER IS THE BLANK CHAR.

EXTERNAL FORM OF OBJECTS (EXAMPLES)

INTEGERS	1 23 -5	
CHARACTER STRINGS	≠ABCDEFGHIJK≠	
BLANK ATOMS	BLK123	NOT READABLE
LABELS	;LABEL45;	NOT READABLE
PROCEDURES	/PROC RANDOM/	NOT READABLE
UNDEFINED VALUE	OM	
EMPTY SET	NL	
EMPTY TUPLE	NULT	
SET	≤1 2 3 4≥	
TUPLE	<1 2 3 4>	
TRUE, FALSE	TRUE FALSE	

REDEFINITION OF THE FILES INFILE AND OUTFILE CAN BE ACHIEVED BY STATEMENTS SUCH AS

```
INFILE = MAKFILE(LOCALFILENAME,LINEWIDTH);
```

WHERE LOCALFILENAME IS A CHARACTER STRING AND
LINEWIDTH IS AN INTEGER.

FOR EXAMPLE, EXECUTION OF OUTFILE=MAKFILE(≠MYOUT≠,120) WILL CAUSE ALL OUTPUT PRODUCED BY SUBSEQUENT WRITSETL STATEMENTS TO BE ROUTED TO THE SCOPE FILE NAMED MYOUT.

2.1.11. BALM4 FUNCTIONS AND OPERATORS

ALL THE BALM4 FUNCTIONS AND OPERATORS ARE AVAILABLE IN BALMSETL, HOWEVER, SOME OPERATORS HAVE NEW MEANINGS IN BALMSETL (SEE SECTION 2.2.), CONSULT THE BALM4 MANUAL FOR DETAILS ABOUT THESE AND OTHER SYNTACTIC ASPECTS OF THE LANGUAGE.

2,1,12, SINISTER CALLS

THE FOLLOWING STATEMENTS REPLACE AN ELEMENT OF A FUNCTION OR A TUPLE BY ANOTHER ELEMENT:

```
LET F OF O1 BE O2
LET F SOF O BE S
LET F OFN LIST(O1,O2,...,ON) BE O
LET F SOFN LIST(O1,O2,...,ON) BE S
LET T COMP I BE O
LET HEAD T BE O
LET TAIL T BE O
```

2.2, MISCELLANEOUS FUNCTIONS, OPERATORS AND VARIABLES

NEXTELT(X,Y)

THIS FUNCTION MAY BE USED TO ITERATE OVER A SET OR A TUPLE X THE AUXILIARY VARIABLE Y MUST BE SET BEFORE THE FIRST CALL TO NEXTELT BY A STATEMENT SUCH AS :

```
Y = COPY(NILVECT)
```

THE FUNCTION NEXTELT RETURNS THEN A NEW ELEMENT OF X AT EVERY CALL, CARE MUST BE TAKEN NOT TO CHANGE THE VALUE OF X RO OF Y BETWEEN CALLS TO NEXTELT,

WHEN THE COLLECTION HAS BEEN EXHAUSTED, NEXTELT WILL RETURN THE UNDEFINED VALUE,

EXAMPLE ; USE OF NEXTELT TO PRINT THE ELEMENTS OF A TUPLE T

```
P=COPY(NILVECT);
Q=NEXTELT(T,P);
WHILE DFD Q REPEAT
  DO PRINT(Q), Q=NEXTELT(T,P) END;
```

O1 EQ O2
O1 NE O2

THESE TWO OPERATORS - NOT TO BE CONFUSED WITH THE CORRESPONDING BALM4 OPERATORS - ARE USED TO TEST FOR EQUALITY IN A MORE RESTRICTED SENSE THAN THE EQUAL AND NEQUAL OPERATORS, O1 EQ O2 IS TRUE (O1 NE O2 IS FALSE) IFF THE INTERNAL REPRESENTATIONS OF O1 AND O2 ARE IDENTICAL, FOR INSTANCE, IF O1 AND O2 ARE SETS, THEIR ELEMENTS MUST BE STORED IN THE SAME ORDER AND THEIR HASHTABLES MUST BE OF EQUAL LENGTHS,

IDENTQ(O1,O2)

THIS FUNCTION MAY BE USED TO CHECK FOR IDENTITY OF O1 AND

O2, IT RETURN TRUE IF O1 EQ O2 IN THE BALM4 SENSE, (SEE THE BALM4 MANUAL).

SAVESETL()

SINCE THE BALM4 SYSTEM IS INCREMENTAL, IT IS POSSIBLE TO EXECUTE PART OF A PROGRAM, TO SAVE THE CURRENT STATUS(INCLUDING STACK,SYMBOL TABLE AND HEAP) ON A FILE AND TO RESUME EXECUTION OF THE NEXT PART OF THE PROGRAM AT A LATER TIME.

THE PROCEDURE SAVESETL() MAY BE USED FOR THAT PURPOSE, WHEN THE PROCEDURE IS CALLED, IT SAVES THE STATUS OF THE SYSTEM ON A LOCAL FILE NAMED TAPE9 , SEE SECTION 4,2, FOR DETAILS CONCERNING THE CATALOGING OF THIS FILE AND THE RESUMPTION OF EXECUTION FROM THE CATALOGED FILE.

LEVEL

THIS IS AN INTEGER VARIABLE (WHOSE INITIAL VALUE IS 10) WHICH FIXES THE DEPTH OF THE PROCEDURE CALL NESTING DISPLAYED WHEN A TERMINAL ERROR IS DETECTED, THIS DEPTH CAN BE CHANGED BY A REGULAR ASSIGNMENT SUCH AS

```
LEVEL=20;
```

CRASHMAX

THIS INTEGER VARIABLE SPECIFIES THE MAXIMUM NUMBER OF CRASHES ALLOWED BEFORE TERMINATION OF A PROGRAM, IT IS INITIALIZED TO 5.

RANDOM(I)

THIS FUNCTION RETURNS A PSEUDO-RANDOM INTEGER IN THE RANGE 1 → I

STRINGOF(O)

THIS FUNCTION RETURNS A CHARACTER STRING WHICH IS THE EXTERNAL REPRESENTATION OF OBJECT O.

PAIRTUP(O)

THIS FUNCTION RETURNS TRUE IFF O IS A TUPLE WITH MORE THAN ONE COMPONENT.

3, PECULIARITIES DUE TO BALM4

3,1, NAMES

3,1,1, IDENTIFIERS

THE NAMES USED AS IDENTIFIERS MAY NOT BE LONGER THAN 8 CHARACTERS,

BECAUSE OF THE STRUCTURE OF THE LANGUAGE AND ITS LESS THAN PERFECT NAME-PROTECTION FEATURE, A LARGE NUMBER OF NAMES MUST NOT BE USED AS IDENTIFIERS OF USER VARIABLES, FAILURE TO RESPECT THIS RULE IN A BALMSETL RUN MAY RESULT IN THE DESTRUCTION OF THE BALMSETL SYSTEM FOR THIS RUN, THE LIST OF RESERVED WORDS CAN BE FOUND IN SECTION 5,

NOTE: THE MORE SOPHISTICATED USER MAY WANT TO ASSIGN NEW VALUES TO RESERVED VARIABLES ANYHOW, THE SYSTEM LISTING OF SECTION 8 SHOULD BE USEFUL FOR THAT PURPOSE,

3,1,2, NAME SCOPING

THE CURRENT VERSION OF BALMSETL IS COMPILED IN SUCH A WAY AS TO USE - AND ALLOW THE USE OF - SO CALLED BALM4

GLOBAL

AND

LOCAL VARIABLES,

THIS FEATURE MAY BE OVERRIDEN TO ALLOW THE USE OF SO CALLED VERY LOCAL VARIABLES
WE DO NOT RECOMMEND THIS, AS IT MAY LEAD TO PROBLEMS IN THE APPLICATION OF MACROS,

BECAUSE OF THE LACK OF CLARITY OF THE BALM4 MANUAL ON THIS SUBJECT, WE GIVE BELOW A SHORT EXPLANATION OF THE NAME SCOPING RULES OF BALMSETL:

A GLOBAL VARIABLE IS ONE WHICH IS NOT DECLARED EXPLICITLY, IT IS KNOWN AT THE OUTERMOST PROGRAM LEVEL AND INSIDE ALL BEGIN-END BLOCKS WHERE THE NAME OF THE GLOBAL VARIABLE HAS NOT BEEN USED AS THE NAME OF A DECLARED LOCAL VARIABLE,

A LOCAL VARIABLE IS DECLARED WITHIN A BEGIN-END BLOCK BY ENTERING ITS NAME - OPTIONALLY PRECEDED BY A $ SIGN - IN A PARENTHESIZED LIST IMMEDIATELY FOLLOWING THE KEYWORD BEGIN, EXAMPLE:

```
BEGIN(A,$B,C),
....,     THE BLOCK CONTAINS 3 LOCAL VARIABLES
....,     NAMED A, B AND C,
END
```

A LOCAL VARIABLE IS KNOWN IN THE BLOCK WHERE IT IS DECLARED AND IN ALL BLOCKS(AND PROCEDURES) WHICH ARE CALLED WITHIN

THAT BLOCK AND WHICH DON≠T CONTAIN A DECLARATION OF A LOCAL VARIABLE OF THE SAME NAME.
EXAMPLE:

```
A=10;       COMMENT≠OUTERMOST PROGRAM LEVEL, A IS GLOBAL≠
B=5;        COMMENT ≠B IS GLOBAL TOO≠
BEGIN(A,C,D),   COMMENT ≠A,C,D ARE LOCAL≠
  A=3,
  C=A,      COMMENT ≠C IS 3≠
  D=7,
  PROC1(A,C),  COMMENT ≠ PROCEDURE CALL ≠
  BEGIN(A,D,E),COMMENT ≠ AN INTERNAL BLOCK≠
    A=B,         COMMENT≠A IS 5≠
    D=A,         COMMENT≠D IS 5≠
    E=C          COMMENT≠E IS 3≠
  END
END;
...
...
PROC1=PROC(X,Y),
        BEGIN(C,D),
          C=A,    COMMENT≠C WILL BE 3 AT THE CALL≠
          D=B     COMMENT≠D WILL BE 5 AT THE CALL≠
        END
      END;
```

3,2, SIMPLE AND COMPOUND OBJECTS

THE FOLLOWING DATA TYPES ARE CALLED SIMPLE:

INTEGER,LABEL,IDENTIFIER,BOOLEAN

THE FOLLOWING DATA TYPES ARE CALLED COMPOUND:

CHARACTER STRING,BLANK ATOM,SET,TUPLE,LIST,
VECTOR,PROCEDURE

THE MAIN DIFFERENCE BETWEEN THESE TWO KINDS OF OBJECTS IS THAT THE VALUES OF SIMPLE TYPES ARE STORED IN THE VARIABLES, WHEREAS THE VALUES OF COMPOUND TYPES ARE STORED IN THE HEAP, THE VARIABLES CONTAIN POINTERS TO THE VALUES INSTEAD OF THE VALUES THEMSELVES.
THIS DISTICTION IS OF IMPORTANCE TO UNDERSTAND THE MANY PROBLEMS RELATED TO THE COPY-RULE(NEXT SECTION).

3,3, ASSIGNMENTS AND COPYING

THE ASSIGNMENT

A = B

COPIES THE CONTENTS OF THE VARIABLE B INTO THE CONTENTS OF VARIABLE A. IF B IS A VARIABLE CONTAINING A SIMPLE DATA ITEM, THEN THE VALUE OF B IS COPIED AND STORED IN A. IF B CONTAINS A COMPOUND DATA ITEM, A POINTER TO THIS DATA ITEM IS COPIED AND STORED IN A. IN SUCH A CASE, A AND B POINT TO THE SAME VALUE IN THE HEAP AND SUBSEQUENT MODIFICATION OF A WILL MODIFY B AND CONVERSELY. IN ORDER TO PREVENT THIS, ONE SHOULD REQUEST A COPY OF THE HEAP

VALUE BY WRITING
```
A = COPY(B)
```

FOR INSTANCE, WHEN A SET S IS INITIALLY EMPTY, BUT WILL BE AUGMENTED LATER ON, ONE SHOULD BE CAREFUL TO INITIALIZE S BY
```
S = COPY(NL)
```

NOTE THAT SOME OPERATIONS ON SETS AND TUPLES WILL PRODUCE COPIES WHILE OTHER OPERATIONS MODIFY THE OPERANDS. FOR INSTANCE:

AUGMENT(S,O)	MODIFIES THE SET S AND RETURNS THE NEW S
S WITH O	RETURNS A MODIFIED COPY OF SET S

3.4. PARAMETERS AND PROCEDURES

THE PROCEDURE MECHANISMS OF BALM4 SPECIFY THAT ALL THE CONTENTS OF THE VARIABLES SERVING AS ACTUAL PARAMETERS IN A PROCEDURE CALL ARE STACKED AT PROCEDURE ENTRY AND RESTORED AT PROCEDURE EXIT TO THE VALUES THEY POSSESED BEFORE THE CALL.
EXAMPLE:

```
PROC1=PROC(X),
          BEGIN(B),
            B=X,
            X=X+3
          END
      END;
...
...
A=5;
PROC1(A);
...
COMMENT=DURING THE CALL TO PROC1, B WILL HAVE THE VALUE 5
    AND X WILL HAVE THE VALUE 8, A WILL STILL BE 5,
```

AS A RESULT OF THIS, IT IS IMPOSSIBLE TO MODIFY THE VALUES OF SIMPLE DATA ITEMS BY A PROCEDURE CALL; HOWEVER, IT IS POSSIBLE TO MODIFY PARTS OF COMPOUND DATA ITEMS, SINCE WHAT IS RESTORED AT PROCEDURE EXIT IS A POINTER TO THE HEAP VALUE; THE LATTER MAY THUS BE CHANGED PERMANENTLY BY A PROCEDURE CALL.
EXAMPLE:

```
A = GENSET(1,2,3);
PROC1(A);
...
...
PROC1=PROC(X),
       AUGMENT(X,4)
       END;
COMMENT ≠ AFTER THE CALL TO PROC1, A IS THE SET WITH 4
          ELEMENTS 1,2,3 AND 4 ≠
```

3.5. BEGIN=END BLOCKS, DO=GROUPS, PROCEDURES

THESE 3 LANGUAGE CONSTRUCTS ARE ALL USED TO GROUP STATEMENTS TOGETHER, IT IS IMPORTANT TO REMEMBER THAT:

- LABELS ARE ALLOWED ONLY ON STATEMENTS WITHIN A BEGIN=END

BLOCK.

- THE RETURN STATEMENT IS ASSOCIATED WITH THE LAST BEGIN-END BLOCK ENTERED DYNAMICALLY WHEN THE RETURN IS EXECUTED IT PRODUCES A RETURN FROM THAT BLOCK. IF A RETURN IS EXECUTED FROM A PROCEDURE WHOSE BODY IS NOT A BEGIN-END BLOCK, THE RETURN WILL FORCE EXIT FROM THE BEGIN-END BLOC IN WHICH THE PROCEDURE WAS CALLED.

- THE BODY OF A PROCEDURE MAY BE
  - A SIMPLE STATEMENT
  - A DO-GROUP
  - A BEGIN-END BLOCK WITH OR WITHOUT LOCAL VARIABLES.

- DON≠T FORGET THE (), AFTER THE KEYWORD BEGIN WHEN THE BLOCK HAS NO LOCAL VARIABLES.

4. OPERATION OF THE SYSTEM

4.1. GENERAL

THE BALMSETL SYSTEM TAKES APPROXIMATELY 61 K(OCTAL) WORDS OF CORE. IN ORDER TO ALLOW THE STACK AND HEAP TO GROW DYNAMICALLY, ONE SHOULD SPECIFY A FIELD LENGTH OF 65 K(OCTAL) MINIMUM.

THE SYSTEM IS COMPOSED OF TWO PARTS, AVAILABLE AS 2 PERMANENT FILES:

1. THE BALM4 INTERPRETER, FILE BLM1208
2. THE BALMSETL TRANSLATOR, FILE SAV1208

THE CURRENT VERSIONS OF THESE TWO FILES WILL ALWAYS BE AVAILABLE AS THE HIGHEST CYCLES OF THE TWO PERMANENT FILES.

THE BALMSETL SYSTEM IS VERY SLOW; BE CAREFUL TO ALLOCATE ENOUGH TIME FOR THE JOB. INTERCOM USERS SHOULD REQUEST A HIGHER-THAN-AVERAGE EXECUTION TIME ALLOCATION.

4.2. BATCH RUNS

THE FOLLOWING DECK MAY BE USED TO RUN BALMSETL ON A SOURCE DECK:

```
ANNNNNN,CM130000,T100, *BALMSETL*
ATTACH(BALM4,BLM1208)
ATTACH(BLM4SVD,SAV1208)
BALM4,
E=0=R

     THE BALMSETL SOURCE DECK

E=0=F
```

THE FOLLOWING DECK MAY BE USED TO RUN BALMSETL ON A SOURCE FILE NAMED, FOR INSTANCE, INFILE

```
ANNNNNN,CM130000,T100, *BALMSETL*
ATTACH(BALM4,BLM1208)
ATTACH(BLM4SVD,SAV1208)
ATTACH(IN,INFILE)
BALM4(IN)
E=0=F
```

IF THE FUNCTION SAVESETL() HAS BEEN USED(SEE SECTION 2), ONE MAY CATALOG THE SAVED BALMSETL SYSTEM BY ADDING THE FOLLOWING CARD AFTER THE BALM4, OR THE BALM4(,IN) CARD:

```
CATALOG(TAPE9,PFNAME,RP=999,PW=PASSWORD)
```

WHERE PFNAME IS A SUITABLE PERMANENT FILE NAME
PASSWORD IS A SUITABLE PASSWORD

TO RESUME EXECUTION FROM A SAVED BALMSETL SYSTEM, ONE MAY USE THE FOLLOWING DECK:

```
ANNNNNN,CM130000,T100, *BALMSETL*
ATTACH(BALM4,BLM1208)
ATTACH(BLM4SVD,PFNAME)
BALM4,
E=O=R

    BALMSETL SOURCE DECK

E=O=F
```

WHERE PFNAME IS THE NAME OF THE PERMANENT FILE CONTAINING THE SAVED BALMSETL SYSTEM.

IF THE SOURCE TEXT COMES FROM A FILE, DO AS ABOVE

```
ATTACH(IN,INFILE)
BALM4(IN)
```

4.3. RUNS UNDER INTERCOM

THESE RUNS ARE SIMILAR TO THE BATCH RUNS; BALM4 HAS NO REAL INTERACTIVE FACILITIES SUCH AS A TEXT EDITOR.

REPLACE THE JOB CARD IN THE BATCH DECK BY THE COMMANDS:

```
CONNECT(INPUT)              USED ONLY IF INPUT IS FROM CONSOLE
EFL(65000)
ETL(100)
```

REMEMBER THAT 60K IS THE MAXIMUM CORE AVAILABLE UNDER INTERCOM. REMEMBER ALSO THAT THE COMMAND ETL(100) WILL BE INEFFECTIVE IF YOUR STANDARD INTERCOM TIME=LIMIT(10 SEC) HAS NOT BEEN CHANGED.

WHEN YOU ENTER THE COMMAND BALM4,, THE SYSTEM WILL RESPOND WITH THE FOLLOWING MESSAGES:

```
MBALM EXECUTION          01/12/72
LOAD FILE EMPTY - RESUME ALL INVOKED
≠BALMSETL SAVED ON TAPE9≠
```

IGNORE ALL THESE MESSAGES. ONLY AFTER THEY HAVE BEEN DISPLAYED IS THE SYSTEM READY TO ACCEPT INPUT TEXT. THE FIRST STATEMENT ENTERED SHOULD NORMALLY BE:

```
TTYFLAG=TRUE;
```

THIS WILL SUPPRESS ANNOYING ECHO≠S.
FROM THIS POINT ONWARDS, THE SYSTEM WILL INDICATE ITS READYNESS TO ACCEPT INPUT BY TYPING A * AT THE LEFT MARGIN.

IF THE INPUT IS TO COME FROM A FILE, DO NOT USE THE COMMAND CONNECT(INPUT) AND REPLACE THE COMMAND BALM4, BY THE COMMANDS:

```
ATTACH(IN,INFILE)
BALM4(IN)
```

5. RESERVED WORDS

THE FOLLOWING IS AN ALPHABETICAL LIST OF THE RESERVED WORDS OF THE BALMSETL SYSTEM. THESE IDENTIFIERS SHOULD NOT BE USED AS NAMES OF USER-CREATED VARIABLES. ASSIGNMENT OF VALUES TO SUCH VARIABLES MAY LEAD TO THE DESTRUCTION OF THE SYSTEM OR TO THE LOSS OF FEATURES. FOR INSTANCE, USE OF THE NAME GENSET FOR A VARIABLE WOULD MEAN THE LOSS OF THE SET-CREATION CAPABILITY OF BALMSETL...

THIS LIST CONTAINS NAMES WHICH ARE NOT TERMINATED BY A DOT AND WHICH ARE THEREFORE ACCESSIBLE TO THE USERS. THESE NAMES ARE USED TO IDENTIFY OPERATORS(U=UNARY,I=INFIX,B=BRACKET), PROCEDURES(P), OTHER GLOBAL VARIABLES(V) AND MACRO KEYWORDS(M) [WHICH ARE, IN FACT, REGULAR OPERATORS].

ABS	U
AND	I
APPEND	P
ARB	U
ARGUMENT	P*
ATOM	U
AUGMENT	P
BACKSPAC	P*
BE	M
BEGIN	B
BLANK	V
BLANKG	P
BOF	I
BOFN	I
BRACKET	P
BREAKUP	P
BSTRG	P
CFROMV	P*
CLOSE	P*
CODEG	P*
COMP	I
COMPILE	P
COMPL	P*
CONCAT	P*
CONCATV	P*
CONSTRUC	P
COPY	P
CRASHMAX	V
DEC	U
DFD	U
DIMINISH	P
DIMF	P
DIMFN	P
DO	B
DSLSH	I
DUMMY	P
DUP	P
EL	I
ELSE	I
ELSEIF	I
END	I
ENDFILE	P*

EQ	I
EQUAL	I
ERROR	P
EXECUTE	P
EXISTS	M
EXPAND	P
FALSE	V
FIRSTWD	P
FOR	U
FORALL	M
FR	M
FROMSET	P
GARBCOLL	P*
GE	I
GENSET	P
GENSYM	P
GENTUP	P
GETPRCP	P*
GETWD	P
GO	U
GOTO	U
GT	I
HD	U
HEAD	U
IDENTC	P
IDFROMC	P*
IDFROMS	P*
IDQ	P*
IF	U
IFROMID	P
IN	M
INCS	I
INDEX	P
INEG	P*
INFIX	P
INR	M
INTERSCT	P
INTQ	P*
IS	M
LAND	P*
LBLQ	P*
LE	I
LENGTH	P
LESF	I
LESFN	I
LESS	I
LET	M
LEVEL	V
LFROMV	P
LIST	P*
LOR	P*
LOGQ	P*
LOOKUP	P
LT	I
MACRO	P
MAKFILE	P
MAKPRCPS	P
MACVAR	V
MAKALCCA	P
MAKVECTO	P*

MAKVLCCA	P
MAPX	P
MAX	I
MEMBER	P
MIN	I
MODE	P*
NE	I
NELT	U
NEXTELT	P
NEQUAL	I
NEWAT	P
NIL	V
NILQ	P*
NILVECT	V
NL	V
NOMEGAP	P
NOT	U
NPOW	P
NULB	V
NULC	V
NULL	U
NULLSET	V
NULT	V
NULLTLPL	V
NUMARGS	P*
OCT	U
OCTMODE	V
OF	I
OFN	I
OMEGAP	P
OPEN	P*
OR	I
ORDINAL	P
PAIRQ	P*
PAIRTUP	P*
PL	U
POW	P
PRINT	P
PROC	B
PROCTRAC	P
PROPL	P*
PROTECT	P*
PRT	P
PRTMAP	P
PTRACE	P
QUOTE	P*
QUANT	M
RANDOM	P
READ	P
RDLINE	P*
REMARK	P*
REMINFIX	P
REMMACRO	P
REMUNARY	P
REPEAT	I
RESTAT	P
RESUMEAL	P*
RETURN	U
REWIND	P*
RPLACA	P*

RPLACD	P*
SAVEALL	P*
SAVEBALM	P
SAVESETL	P
SAVSTAT	P
SETINDEX	P
SETMODE	P*
SETOF	M
SETPRCPL	P*
SETPRCPY	P
SETQQ	P
SETSUB	P*
SETSUBV	P*
SETVALUE	P*
SFROMID	P*
SFROMV	P*
SHIFT	P*
SIM	I
SIZ	P
SIZE	U
SKIPWD	P
SOF	I
SOFN	I
STKTRACE	P*
STRING	P*
STRINGOF	U
STOP	P*
SUB	P*
SUBST	P
SUBV	P*
SUCH	M
SYMDIF	P
SYSLIST	V
TAIL	U
TAILN	I
TAILSPEC	U
TALKATIV	V
THEN	I
TIME	P*
TL	U
TRACE	V
TRANSLAT	P
TRUE	V
TTYFLAG	V
TUPQ	P
TYPE	U
UNARY	P
UNDEF	P
UNDFD	U
UNION	I
VALUE	U
VECTOR	P*
VFROML	P
VFROMS	P*
WHERE	M
WHILE	U
WITH	I
WRLINE	P*
XOR	P*
ZR	U

* A BALM PRIMITIVE WHICH ACTS LIKE A PROCEDURE

6. DIFFERENCES WITH NEWSLETTER 49

THE FOLLOWING FUNCTIONS DESCRIBED IN NEWSLETTER 49 HAVE NOT BEEN IMPLEMENTED:

VALUE	AS
TRIPLE	EXTERNAL-REAL
REPINT	INTERNAL-REAL
REPREAL	SET-AS-TUPLE
REPBSTR	TUPLE-AS-REAL
REPCSTR	TUPLE-AS-BSTRING
REPLABEL	TUPLE-AS-CSTRING
REPBLANK	TUPLE-AS-SET
REPSUBR	NEXTNPOW
REPFUN	RANDOM
ATOMTF	RN
ARITH	RANDOM-INT
QUOTIENT	RANDOM-REAL
TYPE	RANDOM-SET
PAIR	RANDOM-SIMPLY
PMINUS	EXP-II
PPLUS	EXP-RI
FLOOR	EXP-RR
CEILING	SOFN-ARGS-OK
ALL	OF-CSTRING
ANY	OF-BSTRING
ANYS	OFN
EXTERNAL-INT	OFN-ARGS-OK
INTERNAL-INT	BOF-CSTRING
BITR	BOF-BSTRING
BITR-INT	BOFN
BITR-BSTRING	BOFN-ARGS-OK
LE-SET	

THE FOLLOWING BALMSETL FUNCTIONS ARE NOT DESCRIBED IN NEWSLETTER 49:

IDENTQ	PRT
IDENTFIX	SAVESETL
APPEND	CONSTP
STR	MAKTUP
SUBSTR	MAKSETUP
SFROML	MAKOUTUP
SFROMI	CRASH
LFROMS	EACHWITH
IFROMS	GETWD
ATP	SKIPWD
PLUSVD	FIRSTWD
MINSVD	SELTUPL
TIMESVD	PROT
SLHSVD	DUP
PUT	SETSIN
PUTSET	SETTUP
PUTUP	SETFUN
NXTLTT2	SETFUNS
TAILN	TAILSPEC

SETFUNN
HEADSIN
FROMSET
DUPL

SETFUNNS
TAILSIN
ERRFOR
INCS

7. DESCRIPTION OF THE SYSTEM

THE SYSTEM IS COMPOSED OF THE FOLLOWING PARTS:

- FIXES TO BALM4: PROCEDURES IDENT TO SAVESETL.
- BALMSETL PROPER:
  - CONSTANTS
  - GENERAL PROCEDURES TO MANIPULATE SETL OBJECTS
    (UP TO PROCEDURE CHECKFRS).
  - SETL ROUTINES
  - OPERATOR DECLARATIONS
  - MACRO DEFINITIONS.

INDEX TO BALMSETL PROCEDURES

```
SRC   66        DUP,=   PROC(),
SRC 1247        EACHWIT,= PROC(S,X),
SRC  591        ELTF, =   PROC(X,S),
SRC  615        ELTFCST,= PROC(X,S),
SRC  625        ELTFSET,=  PROC(X,S),
SRC  604        ELTUPL,=   PROC( X,S),
SRC  543        EQUALT, =  PROC(A,B),
SRC 1764        ERRFOR, =  PROC(),
SRC   58        ERROR =   PROC(TY),
SRC  432        ERRORIM,= PROC(X),
SRC  438        ERRORTY, = PROC(X),
SRC  445        ERRORVA,= PROC(X),
SRC 1434        FIRSTWD,=  PROC(L,N),
SRC 1753        FROMSET,=  PROC(S),
SRC 1185        GENSET, =  PROC(),
SRC 1194        GENTUP,=   PROC(),
SRC 1418        GETWD,=    PROC(L,N),
SRC  397        HASHCOD,= PROC(X),
SRC  465        HASHCST,= PROC(C),
SRC  473        HASHINT,=  PROC(X),
SRC  479        HASHSTW,= PROC(A,B),
SRC  651        HEAD, =    PROC (TT) ,
SRC 1726        HEADSIN, = PROC(T,X),
SRC   27        IDENTFI,= PROC(A,B),
SRC   24        IDENTQ,=   PROC(A,B),
SRC  123        IFROMS,=   PROC(S,BA),
SRC 1141        INCS, =    PROC(X,Y),
SRC 1109        INTRSCT,= PROC(A,B),
SRC  887        LESF, =    PROC(FF,X),
SRC  926        LESFN, =   PROC(F,X),
SRC  943        LESFNOK, = PROC(F,X),
SRC  720        LESS, =    PROC(S,A),
SRC  900        LESSFOK,=  PROC(G,X),
SRC  120        LFROMS,=   PROC(S),
SRC  486        MAKELAR,= PROC(X),
SRC  503        MAKESML,= PROC(S),
SRC  426        MAKOUTU,= PROC(TUP),
SRC  421        MAKSETU,= PROC(TUP),
SRC  416        MAKTUP,=   PROC(N,L),
SRC 1167        MAX,=      PROC(A,B),
SRC 1172        MIN,=      PROC(A,B),
SRC  151        MINSVD,=   PROC(X,Y),
SRC 1076        MNS,=      PROC(X,Y),
SRC  403        NELT, =    PROC(X),
SRC  587        NEQUALT,=  PROC(A,B),
SRC 1157        NEWAT,=    PROC(),
SRC  528        NEWSMS, =  PROC(N),
SRC 1446        NEXTELT,=  PROC(O,P),
SRC  393        NOMEGAP, = PROC(X),
SRC 1228        NPOW, =    PROC(N,S),
SRC 1459        NXTLTC,=   PROC(O,P),
SRC 1488        NXTLTS,=   PROC(S,P),
SRC 1480        NXTLTT,=   PROC(T,P),
SRC 1469        NXTLTT2, = PROC(T,P),
SRC 1209        OCT,=      PROC(O),
SRC 1356        OF, =      PROC(O,A),
SRC 1405        OFN, =     PROC(F,SBS),
SRC 1367        OFSET, =   PROC(F,X),
SRC 1389        OFTUPL,=   PROC(TUP,B),
SRC  389        OMEGAP,=   PROC(X),
```

```
SRC  988        RAIRTUP,= PROC(X),
SRC 1058        RLS, =    PROC(X,Y),
SRC  148        RLUSVD,=  PROC(X,Y),
SRC 1218        ROW, =    PROC(S),
SRC  323        RRT,=     PROC(A),
SRC  333        RRTMAP =  PROC(S),
SRC  276        RUTBL, =  PROC(L),
SRC  255        RUTITEM, = PROC(L),
SRC  267        RUTLBL, =  PROC(L),
SRC  284        RUTSET, =  PROC(L),
SRC  297        RUTUP, =   PROC(L),
SRC 1262        RANDOM, =  PROC(K),
SRC  200        RDSETL, = PROC(),
SRC   97        REMAIND,= PROC(X,Y),
SRC 1100        REPS,=    PROC(A,B),
SRC  160        SAVSETL,= PROC(),
SRC 1591        SELTUPL, = PROC(T,N),
SRC  815        SETFRML,= PROC(L,N),
SRC 1644        SETFUN, =  PROC(A,B,C),
SRC 1678        SETFUNN, = PROC(A,B,C),
SRC 1657        SETFUNS, = PROC(A,B,C),
SRC 1084        SETMNS,=   PROC(X,Y),
SRC 1885        SETNEST = PROC(X),
SRC  380        SETQQ,=    PROC(X),
SRC 1604        SETSIN, =  PROC(A,B,C),
SRC 1610        SETTUP, =  PROC(A,B,C),
SRC  795        SETWITO, = PROC(A),
SRC  108        SFROMI,=   PROC(I,BA),
SRC  105        SFROML,=   PROC(L),
SRC 1786        SFROMST,= PROC(X),
SRC 1801        SFROMTP,= PROC(X),
SRC 1426        SKIPWD,=   PROC(L,N),
SRC  157        SLHSVD,=   PROC(X,Y),
SRC 1125        SLSH, =    PROC(X,Y),
SRC 1281        SOF, =     PROC(F,X),
SRC 1307        SOFN,=     PROC(F,SBS),
SRC 1351        SOFNERR,=  PROC(),
SRC 1290        SOFOK, =   PROC(F,X),
SRC  377        SPCTUPQ,= PROC(X),
SRC 1698        STFUNNS,=PROC(A,B,C),
SRC   96        STR,=      PROC(A,B), CONCAT(A,B) END;
SRC 1769        STRINGF,= PROC(X),
SRC  690        SUBSTR =  PROC(A,B,C),
SRC  212        SWITCHP,= PROC(),
SRC 1135        SYMDIF,=   PROC(A,B),
SRC  657        TAIL,=     PROC(TT),
SRC  673        TAILN, =   PROC(TT,N),
SRC 1741        TAILSIN,=  PROC(T,X),
SRC  976        TAILSPC,= PROC(T),
SRC  154        TIMESVD,=  PROC(X,Y),
SRC 1094        TMS,=      PROC(A,B),
SRC 1068        TPLUS,=    PROC(T1,T2),
SRC  373        TUPQ,=     PROC(X),
SRC  810        TUPWITO, = PROC(A),
SRC  364        TYPE, =    PROC(X),
SRC 1072        UNION,=    PROC(A,B),
SRC  717        WITH,=     PROC(S,A),
SRC  309        WRITSETL= PROC(),
```

```
                                                                                SRC     2
                                                                                SRC     3
                                                                                SRC     4
COMMENT
        ≠ THE PROGRAM DOES NOT CONTAIN MANY COMMENTS, FOR MORE DETAILS,         SRC     5
          SEE NEWSLETTER 49.....                                                SRC     6
                                                                                SRC     7
≠                                                                               SRC     8
COMMENT                                                                         SRC     9
≠ THE FOLLOWING ROUTINES ARE BALM4 FIXES ≠                                      SRC    10
MAKALOCA(TRUE); COMMENT ≠ TO ADJUST SCOPE OF, PARAMETERS ≠                      SRC    11
MAKVLOCA(TRUE);                                                                 DEC72   1
STRING=PROC(X), STRING(X) END;                                                  SRC    12
COMMENT ≠MAKE , INTO A LETTER≠                                                  SRC    13
CHGCHAR(≡,,10);                                                                 SRC    14
COMMENT ≠ ERASE BALM INITIALIZATION ROUTINES TO SAVE MEMORY ≠                   SRC    15
DO INITIAT,=0,INITIO,=0,INITUNA,=0,INITINF,=0,INITEXP,=0,INITCOD,=0,            SRC    16
   INITORL,=0, INITMIS,=0 END;                                                  SRC    17
COMMENT ≠ SOME MACROS FOR FREQUENTLY USED OPERATIONS ..... ≠                    SRC    18
COMMENT ≠ NUMBER OF ELEMENTS IN A SET OR A TUPLE ≠                              SRC    19
NELTS,(X1) MEANS X1[3][1] ;                                                     SRC    20
TUPLE,(X1) MEANS X1[3][2] ;                                                     SRC    21
HTSIZE,(X1) MEANS X1[3][2] ;                                                    SRC    22
HTLOAD,(X1) MEANS X1[3][3] ;                                                    SRC    23
HASHTAB,(X1) MEANS X1[3][4] ;                                                   SRC    24
IDENTQ,=   PROC(A,B),                                                           SRC    25
             A EQ B                                                             SRC    26
           END;                                                                 SRC    27
IDENTFI,= PROC(A,B),                                                            SRC    28
            BEGIN(I),                                                           SRC    29
              IF IDENTQ,(A,B) THEN RETURN(TRUE),                                SRC    30
              IF NOT (A SIM B) THEN RETURN(NIL),                                SRC    31
              IF STRQ(A) THEN RETURN(EQSTR(A,B)),                               SRC    32
              IF VECTQ(A) THEN DO                                               SRC    33
                 IF SIZE(A) NE SIZE(B) THEN RETURN(NIL),                        SRC    34
                 FOR I=(1,SIZE(A))REPEAT IF NOT IDENTFI,(A[I],B[I])             SRC    35
                 THEN                                                           SRC    36
                 RETURN(NIL), RETURN(TRUE)                                      SRC    37
              END,                                                              SRC    38
              IF PAIRQ(A) THEN IF NOT IDENTFI,(HD A,HD B) THEN RETURN           SRC    39
                (NIL)                                                           SRC    40
              ELSEIF NOT IDENTFI,(TL A,TL B) THEN RETURN(NIL) ELSE              SRC    41
              RETURN(TRUE),                                                     SRC    42
              RETURN(NIL)                                                       SRC    43
            END                                                                 SRC    44
          END;
*****         FIX GETLIST AND GETV TO ACCEPT NL, AS NL AND NULT,                SEPT73  1
*****           AS NULT AND OM, AS OM                                           SEPT73  2
 GETLIST,=PROC(),                                                               SEPT73  3
       BEGIN(ITM,X,Y),                                                          SEPT73  4
         X=Y=NIL:NIL,                                                           SEPT73  5
 MOR,    ITM=RDITEM,(),                                                         SEPT73  6
         IF ITM EQ RPVAR, THEN RETURN TL Y                                      SEPT73  7
         ELSEIF ITM EQ PERVAR, THEN                                             SEPT73  8
           (IF HD X EQ =OM OR HD X EQ =NL OR HD X EQ=NULT THEN GO MOR           SEPT73  9
             ELSE TL X=ITM=RDITEM,())                                           SEPT7310
         ELSEIF ITM EQ IEOS, THEN DO                                            SEPT7311
           NEXT=NEXT-1,                                                         SEPT7312
           PRINT,(LIST(=PARENS,=DONT,=MATCH )), RETURN TL Y                     SEPT7313
           END                                                                  SEPT7314
         ELSE X=ADDON,(X,ITM),                                                  SEPT7315
         GOTO MOR                                                               SEPT7316
```

```
   END END;                                                                    SEPT7317
GETV,=PROC(),                                                                  SEPT7318
    BEGIN(ITM,X,Y),                                                            SEPT7319
        X=Y=NIL;NIL,                                                           SEPT7320
MOR,    ITM=RDITEM,(),                                                         SEPT7321
        IF ITM EQ RBVAR, THEN RETURN TL Y                                      SEPT7322
        ELSEIF ITM EQ PERVAR, AND ( HD X EQ =NL OR HD X EQ =NULT OR            SEPT7323
              HD X EQ =OM) THEN GO MOR                                         SEPT7324
        ELSEIF ITM EQ IEOS, THEN DO                                            SEPT7325
          NEXT=NEXT-1,                                                         SEPT7326
          PRINT,(LIST(=BRACKETS,=DONT,=MATCH)), RETURN TL Y                    SEPT7327
        END                                                                    SEPT7328
        ELSE X=ADDON,(X,ITM),                                                  SEPT7329
        GOTO MOR                                                               SEPT7330
   END END;                                                                    SEPT7331
INFIX(=EQ,1400,1400,=IDENTFI,);                                                SRC    45
COPY =    PROC(X),                                                             SRC    46
            IF X EQ NL, THEN VECTOR(SET,,HASHNUL,,VECTOR(0,0,0,0))             SRC    47
            ELSEIF X EQ NULT, THEN VECTOR(TUPL,,31416,VECTOR(0,NIL))           SRC    48
            ELSEIF VECTQ(X) THEN BEGIN(V,L,I),                                 SRC    49
               L=SIZE X, V=MAKVECTO(L),                                        SRC    50
               FOR I=(1,L) REPEAT V[I] = COPY(X[I]), RETURN(V)                 SRC    51
            END                                                                SRC    52
            ELSEIF PAIRQ(X) THEN COPY(HD X);COPY(TL X)                         SRC    53
            ELSEIF STRQ(X) THEN SUB(X,1,SIZE(X))                               SRC    54
            ELSE X                                                             SRC    55
          END;                                                                 SRC    56
COMMENT ≠ PROCEDURE DUP, TO  CREATE NON-DOTTED NAMES FOR                       SRC    64
           USER LEVEL.........≠                                                SRC    65
DUP,=   PROC(),                                                                SRC    66
          BEGIN(I,ID,SID),                                                     SRC    67
             FOR I=(1,NUMARGS()) REPEAT DO                                     SRC    68
                ID=ARGUMENT(I), SID=SFROMID(ID),                               SRC    69
                SETVALUE(IDFROMS(SUB(SID,1,SIZE(SID)-1)),VALUE ID)             SRC    70
             END                                                               SRC    71
          END                                                                  SRC    72
       END;                                                                    SRC    73
APPEND,=  PROC(X,Y),                                                           SRC    74
            BEGIN(L),                                                          SRC    75
               X=COPY(X), Y=COPY(Y),                                           SRC    76
               IF X EQ NIL THEN DO                                             SRC    77
                  IF Y EQ NIL THEN RETURN(NIL),                                SRC    78
                  IF NOT PAIRQ(Y) THEN RETURN(LIST(Y)),                        SRC    79
                  RETURN(Y)                                                    SRC    80
               END,                                                            SRC    81
               IF NOT PAIRQ(X) THEN DO                                         SRC    82
                  IF Y EQ NIL THEN RETURN(LIST(X)),                            SRC    83
                  IF NOT PAIRQ(Y) THEN RETURN(LIST(X,Y)),                      SRC    84
                  RETURN(X:Y)                                                  SRC    85
               END,                                                            SRC    86
               IF Y EQ NIL THEN RETURN(X),                                     SRC    87
               IF NOT PAIRQ(Y) THEN Y=LIST(Y),                                 SRC    88
               L=X,                                                            SRC    89
               WHILE TL L NE NIL REPEAT L=TL L,                                SRC    90
               TL L=Y,                                                         SRC    91
               RETURN(X)                                                       SRC    92
            END                                                                SRC    93
          END;                                                                 SRC    94
COMMENT ≠ SYNONYMS FOR BALM4 BUILT-IN FUNCTIONS ≠                              SRC    95
STR,=      PROC(A,B), CONCAT(A,B) END;                                         SRC    96
```

```
REMAIND,= PROC(X,Y),                                                    SRC   97
             X=Y*(X/Y)                                                  SRC   98
          END;                                                          SRC   99
DIVIDED,=  PROC(X,Y),                                                   SRC  100
             BEGIN(Q),                                                  SRC  101
                Q=X/Y,RETURN(LIST(Q,X-Y*Q))                             SRC  102
             END                                                        SRC  103
          END ;                                                         SRC  104
SFROML,=   PROC(L),                                                     SRC  105
             SFROMV(VFROML(L))                                          SRC  106
          END;                                                          SRC  107
SFROMI,=   PROC(I,BA),                                                  SRC  108
             BEGIN(L,M,R),                                              SRC  109
                L=NIL,M=I LT 0,IF M THEN I=-I,                          SRC  110
                IF I EQ 0 THEN RETURN(≠0≠),                             SRC  111
                WHILE I NE 0 REPEAT DO                                  SRC  112
                   R=I-(BA*(I/BA))*27, I=I/BA,                          SRC  113
                   L=R;L                                                SRC  114
                END,                                                    SRC  115
                IF M THEN L=38;L,                                       SRC  116
                RETURN(SFROML,(L))                                      SRC  117
             END                                                        SRC  118
          END;                                                          SRC  119
LFROMS,=   PROC(S),                                                     SRC  120
             LFROMV(VFROMS(S))                                          SRC  121
          END;                                                          SRC  122
IFROMS,=   PROC(S,BA),                                                  SRC  123
             BEGIN(K,M,V,I,J),                                          SRC  124
                V=VFROMS(S),                                            SRC  125
                K=1,I=0,                                                SRC  126
                WHILE V[K] EQ 45 REPEAT K=K+1,                          SRC  127
                IF V[K] EQ 37 THEN DO                                   SRC  128
                   K=K+1, M=1                                           SRC  129
                END                                                     SRC  130
                ELSEIF V[K] EQ 38 THEN DO                               SRC  131
                   K=K+1, M=-1                                          SRC  132
                END                                                     SRC  133
                ELSE M=1,                                               SRC  134
                FOR J=(K,SIZE(V))REPEAT                                 SRC  135
                   IF V[J] GE (BA+27) THEN ERRORVA,(                    SRC  136
                   ≠CONVERSION FROM STRING TO INTEGER:                  SRC  137
                   ONE CHARACTER GREATER THAN THE BASE≠)                SRC  138
                   ELSE I=I*BA+V[J]-27,                                 SRC  139
                RETURN(M*I)                                             SRC  140
             END                                                        SRC  141
          END;                                                          SRC  142
ATP,=      PROC(O),                                                     SRC  143
             INTQ(O) OR LOGQ(O) OR CODEQ(O)                             SRC  144
          END;                                                          SRC  145
                                                                        SRC  146
COMMENT ≠ SAVE ORIGINAL ARITHMETIC OPERATIONS≠                          SRC  147
PLUSVD,=   PROC(X,Y),                                                   SRC  148
             X+Y                                                        SRC  149
          END;                                                          SRC  150
MINSVD,=   PROC(X,Y),                                                   SRC  151
             X-Y                                                        SRC  152
          END;                                                          SRC  153
TIMESVD,=  PROC(X,Y),                                                   SRC  154
             X*Y                                                        SRC  155
          END;                                                          SRC  156
```

```
SLHSVD,=   PROC(X,Y),                                                  SRC  157
             X/Y                                                       SRC  158
           END;                                                        SRC  159
COMMENT  ≠SOME SETL CONSTANTS  ..... ≠                                 SRC  160
                                                                       SRC  170
BLANK,,=6;SET,=10;TUPL,=11;                                            SRC  171
HASHNUL,=12345;                                                        SRC  172
MODULUS,=131071;                                                       SRC  173
CRASHMAX=5; CRSHCNT,=0;                                                SRC  174
NEWATNR,=0;                                                            SRC  175
HNC,=27182; HNB,=66256;                                                SRC  176
UNDEF, = VECTOR (0,0,0);                                               SRC  177
NULLSET, = VECTOR(SET,,HASHNUL,,VECTOR(0,0,0,0));                      SRC  178
NL, = NULLSET,;   NL = NL,;                                            SRC  179
NULT,=VECTOR(TUPL,,31416,VECTOR(0,NIL));  NULT = NULT,;                SRC  180
NULLTUP,=NULT,;                                                        SRC  181
NULC=≠≠;                                                               SRC  182
NULB,=LIST(0B);                                                        SRC  183
NILVECT,==[NIL];                                                       SRC  184
BLANKZZ,=BLANK,,;SETZZ,=SET,;TUPLZZ,=TUPL,;                            SRC  185
INTZZ,=1;SUBRZZ,=2;LABZZ,=3;STRZZ,=4;BITSZZ,=5;                        SRC  186
COMMENT ≠ BALMSETL I/O ..........  APRIL 19, 1972 ≠                    SRC  187
                                                                       SRC  188
COMMENT ≠ OLD AND NEW PARAENTHESES ......≠                             SRC  189
DO                                                                     SRC  190
LPVAROZ,=LPVAR,,LBVAROZ,=LBVAR,,RPVAROZ,=RPVAR,,RBVAROZ,=RBVAR,,       SRC  191
LPVARNZ,=IDFROMS(SFROMV(VECTOR(74B)));                                 SRC  192
LBVARNZ,=IDFROMS(SFROMV(VECTOR(72B)));                                 SRC  193
RPVARNZ,=IDFROMS(SFROMV(VECTOR(75B)));                                 SRC  194
RBVARNZ,=IDFROMS(SFROMV(VECTOR(73B)));                                 SRC  195
QSIGN,=EQSIGN,                                                         SRC  196
END;                                                                   SRC  197
INFILE=INPUT; OUTFILE=OUTPUT;                                          SRC  198
                                                                       SRC  199
RDSETL, = PROC(),                                                      SRC  200
            BEGIN(X),                                                  SRC  201
               SWITCHP,(),                                             SEPT7332
               IF (X=READ(INFILE)) EQ =, THEN X=READ(INFILE);          SEPT7333
               SWITCHP,(),                                             SEPT7334
               RETURN(CONVERT,(X))                                     SRC  203
            END                                                        SRC  204
         END;                                                          SRC  205
                                                                       SRC  206
READZZZ(X1) MEANS X1=RDSETL,();                                        SRC  207
READZZZ(X1,X2) MEANS DO READXXX,(X1,X2) END;                           SRC  208
READXXX,(X1) MEANS X1=RDSETL,();                                       SRC  209
READXXX,(X1,X2) MEANS X1=RDSETL,(),READXXX,(X2);                       SRC  210
                                                                       SRC  211
READSETL = PROC(FIL),                                                  JUL73  1
       BEGIN(X),                                                       JUL73  2
          SWITCHP,(),                                                  OCT73  1
          IF (X=READ(FIL)) EQ =, THEN X=READ(FIL),                     OCT73  2
          SWITCHP,(),                                                  OCT73  3
          RETURN(CONVERT,(X))                                          JUL73  4
   END END;                                                            JUL73  5
                                                                       JUL73  6
SWITCHP,= PROC(),                                                      SRC  212
            IF LPVAR, EQ LPVAROZ, THEN DO                              SR   213
               LPVAR,=LPVARNZ,,LBVAR,=LBVARNZ,,                        SRC  214
               RPVAR,=RPVARNZ,,RBVAR,=RBVARNZ,                         SRC  215
```

```
            END ELSE DO                                                    SRC  216
               LPVAR,=LPVAROZ,,LBVAR,=LBVAROZ,,                            SRC  217
               RPVAR,=RPVAROZ,,RBVAR,=RBVAROZ,                             SRC  218
            END                                                            SRC  219
         END;                                                              SRC  220
                                                                           SRC  221
                                                                           SRC  222
CONVERT, = PROC(X),                                                        SRC  223
            BEGIN(Y,Z,I),                                                  SRC  224
               IF INTQ(X) THEN RETURN(X)                                   SRC  225
               ELSEIF STRQ(X) THEN RETURN(X)                               SRC  226
               ELSEIF PAIRQ(X) THEN DO                                     SRC  227
                  Y=COPY(NL,),                                             SRC  228
                  WHILE X NE NIL REPEAT DO                                 SRC  229
                     Z=CONVERT,(HD X),                                     SRC  230
                     IF Z NE UNDEF, THEN AUGMENT,(Y,Z),                    SRC  231
                     X=TL X                                                SRC  232
                  END,                                                     SRC  233
                  RETURN(Y)                                                SRC  234
               END                                                         SRC  235
               ELSEIF VECTQ(X) THEN DO                                     SRC  236
                  Z=SIZE(X),                                               SRC  237
                  IF Z EQ 0 THEN RETURN(COPY(NULT,)),                      SRC  238
                  Y=NIL,                                                   SRC  239
                  FOR I=(Z,1,-1) REPEAT Y=CONVERT,(X[I]);Y,                SRC  240
                  RETURN(VECTOR(TUPL,,HASHCOD,(HD Y),VECTOR(Z,Y)))         SRC  241
               END                                                         SRC  242
               ELSEIF X EQ =OM THEN RETURN(COPY(UNDEF,))                   SRC  243
               ELSEIF X EQ =NL THEN RETURN(COPY(NL,))                      SRC  244
               ELSEIF X EQ =NULT THEN RETURN(COPY(NULT,))                  SRC  245
               ELSEIF X EQ =FALSE THEN RETURN(FALSE)                       SRC  246
               ELSEIF X EQ NDF, THEN RETURN X                              JAN73  1
               ELSEIF IDQ(X) THEN RETURN SFROMID(X)                        NOV72  1
               ELSE RETURN(X)                                              SRC  247
            END                                                            SRC  248
         END;                                                              SRC  249
                                                                           SRC  250
                                                                           SRC  251
PUT,=PUTITEM,; COMMENT ≠ SAVE OLD PUTITEM, ≠                               SRC  252
COLOZZZ,=63B;                                                              SRC  253
                                                                           SRC  254
PUTITEM, = PROC(L),                                                        SRC  255
            IF L EQ NIL THEN PUTCHV,(VFROMS(SFROMID(=FALSE)))              SRC  256
            ELSEIF L EQ NL, THEN PUTCHV,(VFROMS(SFROMID(=NL,)))            DEC72  2
            ELSEIF L EQ NULT, THEN PUTCHV,(VFROMS(SFROMID(=NULT,)))        DEC72  3
            ELSEIF SETQQ,(L) THEN PUTSET,(L)                               SRC  259
            ELSEIF TUPQ,(L) THEN PUTUP,(L)                                 SRC  260
            ELSEIF LBLQ(L) THEN PUTLBL,(L)                                 SRC  261
            ELSEIF L EQ UNDEF, THEN PUTCHV,(VFROMS(SFROMID(=OM,)))         DEC72  4
            ELSEIF BLANKQ,(L) THEN PUTBL,(L)                               SRC  263
            ELSE PUT,(L)                                                   SRC  264
         END;                                                              SRC  265
                                                                           SRC  266
PUTLBL, =  PROC(L),                                                        SRC  267
            BEGIN(),                                                       SRC  268
               IF NEXT GT LLEN-10 THEN TERMLINE(),                         SRC  269
               PUTCH,(COLOZZZ,),PUTCHV,(VFROMS(SFROMID(=LABEL))),          SRC  270
               NEXT=NEXT-1,PUTINT,(L),NEXT=NEXT-1,PUTCH,(COLOZZZ,),        SRC  271
               PUTBLAN,()                                                  SRC  272
            END                                                            SRC  273
```

```
          END;                                                                  SRC  274
                                                                                SRC  275
PUTBL, =    PROC(L),                                                          SRC  276
              BEGIN(),                                                          SRC  277
                 IF NEXT GT LLEN-10 THEN TERMLINE(),                            SRC  278
                 PUTCHV,(VFROMS(SFROMID(=BLK))),                                SRC  279
                 NEXT=NEXT-1,PUTINT,(L[3]),PUTBLAN,()                           SRC  280
              END                                                               SRC  281
           END;                                                                 SRC  282
                                                                                SRC  283
PUTSET, =   PROC(L),                                                          SRC  284
              BEGIN(X,P),                                                       SRC  285
                 IF NEXT GT LLEN-10 THEN TERMLINE(),                            SRC  286
                 PUTCH,(74B),BPCNT=BPCNT+1,                                     SRC  287
                 P=COPY(NILVECT,),X=NEXTELT,(L,P),                              SRC  288
                 WHILE X NE UNDEF, REPEAT DO                                    SRC  289
                    PUTITEM,(X), X=NEXTELT,(L,P)                                SRC  290
                 END,                                                           SRC  291
                 NEXT=NEXT-1,                                                   SRC  292
                 PUTCH,(75B),BPCNT=BPCNT-1,PUTBLAN,()                           SRC  293
              END                                                               SRC  294
           END;                                                                 SRC  295
                                                                                SRC  296
PUTUP, =    PROC(L),                                                          SRC  297
              BEGIN(X),                                                         SRC  298
                 IF NEXT GT LLEN-10 THEN TERMLINE(),                            SRC  299
                 PUTCH,(72B),BPCNT=BPCNT+1,                                     SRC  300
                 X=TUPLE,(L),                                                   SRC  301
                 WHILE X NE NIL REPEAT DO                                       SRC  302
                    PUTITEM,(HD X), X=TL X                                      SRC  303
                 END,                                                           SRC  304
                 NEXT=NEXT-1,PUTCH,(73B),BPCNT=BPCNT-1,PUTBLAN,()               SRC  305
              END                                                               SRC  306
           END;                                                                 SRC  307
                                                                                SRC  308
WRITSETL = PROC(OUTFILE),                                                       JUL73  7
             BEGIN($FN,$LIN,$LLEN,$NEXT,$BPCNT,$TERMLINE,I,N,TR,FIL),           SRC  310
                 TR=TRACE,TRACE=0,N=NUMARGS(),FIL=OUTFILE,                      SRC  311
                 TERMLINE=WRITOUT,,FN=FIL[1],LIN=FIL[5],                        SRC  312
                 LLEN=FIL[6],NEXT=FIL[7],BPCNT=0,                               SRC  313
                 FOR I = (2,N) REPEAT                                           JUL73  8
                     PUTITEM, (ARGUMENT(I)),                                    JUL73  9
                 TERMLINE(),FIL[5]=LIN,FIL[7]=NEXT,TRACE=TR,                    SRC  315
                 RETURN(ARGUMENT(N))                                            SRC  316
              END                                                               SRC  317
           END;                                                                 SRC  318
                                                                                SRC  319
                                                                                SRC  320
                                                                                SRC  321
COMMENT ≠ PRT, PRINTS IN THE OLD FASHION ....! ≠                                SRC  322
PRT,=       PROC(A),                                                            SRC  323
              BEGIN(P,I),                                                       SRC  324
                 P=PUTITEM,,                                                    SRC  325
                 DO PUTITEM,=PUT, END,                                          SRC  326
                 FOR I=(1,NUMARGS()) REPEAT PRINT(ARGUMENT(I));                 SRC  327
                 DO PUTITEM,=P END                                              SRC  328
              END                                                               SRC  329
           END;                                                                 SRC  330
COMMENT ≠ PRTMAP PRINTS A SET, OF, TUPLES AS A MAP ≠                            SRC  331
                                                                                SRC  332
```

```
PRTMAP = PROC(S),                                                    SRC  333
         BEGIN(X,Y,P),                                               SRC  334
            IF NOT SETQQ,(S) THEN ERRORTY,(                          SRC  335
            ≠PRTMAP(X), X NOT A SET,≠),                              SRC  336
            IF S EQ NL, THEN RETURN(PRINT(S)),                       SRC  337
            P=COPY(NILVECT,),                                        SRC  338
            X=NEXTELT,(S,P),                                         SRC  339
            WHILE X NE UNDEF, REPEAT DO                              SRC  340
               IF TUPQ,(X) AND NELTS,(X) GE 2 THEN DO                SRC  341
                  Y=TAIL,(X), IF NELTS,(Y) LE 1 THEN Y=HEAD,(Y),     SRC  342
                  PRINT(HEAD,(X),==,==,==>,Y)                        SRC  343
               END                                                   SRC  344
               ELSE PRINT(X),                                        SRC  345
               X=NEXTELT,(S,P)                                       SRC  346
            END                                                      SRC  347
         END;                                                        SRC  348
UNARY(=VAR,1300,=ATP,);                                              SRC  349
INFIX(=MOD,1450,1450,=REMAIND,);                                     SRC  350
INFIX(=REM,800,799,=REMAIND,);                                       SRC  351
INFIX(=DIVIDE,1601,1600,=DIVIDED,);                                  SRC  352
                                                                     SRC  353
                                                                     SRC  354
COMMENT                                                              SRC  355
≠ BALMSETL ROUTINES START HERE ≠                                     SRC  356
COMMENT                                                              SRC  357
≠ THE ROUTINES WHICH FOLLOW APPEAR MORE OR LESS, IN THE SAME ORDER   SRC  358
  AS IN NEWSLETTER 49, WHICH IS TO BE CONSULTED FOR DETAILED         SRC  359
  COMMENTS ≠                                                         SRC  360
                                                                     SRC  361
                                                                     SRC  362
TYPE, =    PROC(X),                                                  SRC  363
             IF INTQ(X) THEN INTZZ,                                  SRC  364
             ELSEIF VECTQ(X) THEN X[1]                               SRC  365
             ELSEIF STRQ(X) THEN STRZZ,                              SRC  366
             ELSEIF CODEQ(X) THEN SUBRZZ,                            SRC  367
             ELSEIF LBLQ(X) THEN LABZZ,                              SRC  368
             ELSEIF LOGQ(X) THEN BITSZZ,                             SRC  369
             ELSE ERRORTY,(≠TYPE,(X), X NOT A LEGAL OBJECT≠)         SRC  370
           END;                                                      SRC  371
TUPQ,=      PROC(X),                                                 SRC  372
             VECTQ(X) AND IDENTQ,(X[1],TUPL,)                        SRC  373
           END;                                                      SRC  374
COMMENT ≠ PREDICATE FOR TUPLE, OF, LENGTH GREATER THAN 2 ≠           SRC  375
SPCTUPQ,= PROC(X),                                                   SRC  376
             VECTQ(X) AND IDENTQ,(X[1],TUPL,) AND X[3][1] GE 3       SRC  377
           END;                                                      SRC  378
SETQQ,=     PROC(X),                                                 SRC  379
             VECTQ(X) AND IDENTQ,(X[1],SET,)                         SRC  380
           END;                                                      SRC  381
BLANKQ, =   PROC(O),                                                 SRC  382
             VECTQ(O) AND IDENTQ,(O[1],BLANK,)                       SRC  383
           END;                                                      SRC  384
BSTRQ,=     PROC(O),                                                 SRC  385
             IF PAIRQ(O) THEN INTQ(HD O) ELSE FALSE                  SRC  386
           END;                                                      SRC  387
OMEGAP,=    PROC(X),                                                 SRC  388
             IF X EQ UNDEF, THEN                                     SRC  389
             TRUE ELSE FALSE                                         SRC  390
           END;                                                      SRC  391
                                                                     SRC  392
```

```
NOMEGAP, = PROC(X),                                                    SRC  393
             IF X EQ UNDEF, THEN                                       SRC  394
             FALSE ELSE TRUE                                           SRC  395
          END;                                                         SRC  396
HASHCOD,= PROC(X),                                                     SRC  397
             IF VECTQ(X)THEN X[2]                                      SRC  398
             ELSEIF INTQ(X)THEN HASHINT,(X)ELSEIF STRQ(X) THEN         SRC  399
               HASHCST,(X)                                             SRC  400
             ELSEIF BSTRQ,(X) THEN HASHBSTR(X) ELSE 17                 SRC  401
          END;                                                         SRC  402
NELT, =    PROC(X),                                                    SRC  403
             IF STRQ(X) THEN SIZE(X)                                   SRC  404
             ELSEIF SETQQ,(X) OR TUPQ,(X) THEN X[3][1]                 SRC  405
             ELSEIF ATOM,(X) THEN 1                                    SRC  406
             ELSEIF BSTRQ,(X) THEN BLEN(X)                             SRC  407
             ELSEIF X EQ UNDEF, THEN ERRORVA,(                         SRC  408
             ≠NELT, X,X IS UNDEFINED≠)                                 SRC  409
             ELSE ERRORTY,(≠NELT, X, X NOT A LEGAL TYPE;≠)             SRC  410
          END;                                                         SRC  411
CONSTP,=   PROC(O,T),                                                  SRC  412
             VECTOR(TUPL,,HASHCOD,(O),VECTOR(NELTS,(T)+1,O;TUPLE,(T))) SRC  413
             END;                                                      SRC  414
COMMENT ≠ CREATE TUPLE, FROM A LIST L OF, N ELEMENTS ≠                 SRC  415
MAKTUP,=   PROC(N,L),                                                  SRC  416
             VECTOR(TUPL,,HASHCOD,(HD L), VECTOR(N,L))                 SRC  417
          END;                                                         SRC  418
COMMENT ≠ TUPLES INSIDE SETS ARE REPRESENTED AS TWO= OR THREE=         SRC  419
         VECTORS,OUTSIDE SETS AS LISTS,HENCE TWO CONVERSION ROUTINES ≠ SRC  420
                                                                       SRC  421
MAKSETU,= PROC(TUP),                                                   SRC  422
             IF NOT TUPQ,(TUP) THEN TUP ELSEIF PAIRQ(TUP[3][2])THEN    SRC  423
             VECTOR(TUP[1],TUP[2],VECTOR(TUP[3][1],VFROML(TUP[3][2]))) SRC  424
             ELSE TUP                                                  SRC  425
          END;                                                         SRC  426
MAKOUTU,= PROC(TUP),                                                   SRC  427
             IF NOT TUPQ,(TUP) THEN TUP ELSEIF VECTQ(TUP[3][2]) THEN   SRC  428
             VECTOR(TUP[1],TUP[2],VECTOR(TUP[3][1],LFROMV(TUP[3][2]))) SRC  429
             ELSE TUP                                                  SRC  430
          END;                                                         SRC  431
COMMENT ≠ ERROR ROUTINES ≠                                             SRC  432
ERRORIM,= PROC(X),                                                     SRC  433
             BEGIN(),                                                  SRC  434
                PRINT(≠IMPLEMENTATION ERROR:≠,X),                      SRC  435
                CRASH,()                                               SRC  436
             END                                                       SRC  437
          END; COMMENT ≠CRASH, IS RECOVERY PROCEDURE ≠                 SRC  438
ERRORTY, = PROC(X),                                                    SRC  439
             BEGIN(),                                                  SRC  440
                PRINT(                                                 SRC  441
                ≠INVALID DATA TYPE, FOR THE SETL OPERATION: ≠,X),      SRC  442
                CRASH,()                                               SRC  443
             END                                                       SRC  444
          END;                                                         SRC  445
ERRORVA,= PROC(X),                                                     SRC  446
             BEGIN(),                                                  SRC  447
               PRINT(≠RUN-TIME VALUE ERROR IN ≠,X),                    SRC  448
               CRASH,()                                                SRC  449
             END                                                       SRC  450
          END;                                                         SRC  462

COMMENT ≠ HASH-RELATED ROUTINES  ≠                                     SRC  463
```

```
                                                                                SRC  464
                                                                                SRC  465
HASHCST,= PROC(C),                                                              SRC  466
            BEGIN(I1,I,V,K),                                                    SRC  467
               I1=HNC,, K=MIN,(SIZE(C),8),                                      SRC  468
               V=VFROMS(SUB(C,1,K)),                                            SRC  469
               FOR I=(1,K)REPEAT I1=(I1+V[I]*(5↑I))REM MODULUS,,                SRC  470
               RETURN(IF I1 GE 0 THEN I1 ELSE -I1)                              SRC  471
            END                                                                 SRC  472
         END;                                                                   SRC  473
HASHINT,=  PROC(X),                                                             SRC  474
            IF X GE 0 THEN X ELSE -X                                            SRC  475
         END;                                                                   SRC  476
COMPRHS, = PROC(MAXVALUE,HASHCOD,),                                             SRC  477
            (HASHCOD, REM MAXVALUE)+1                                           SRC  478
         END;                                                                   SRC  479
HASHSTW,= PROC(A,B),                                                            SRC  480
            XOR(HASHCOD,(A),HASHCOD,(B))                                        SRC  481
         END;                                                                   SRC  482
HASHSTL, = HASHSTW,;                                                            SRC  483
MAXDEN, =2;                                                                     SRC  484
SMLHSTA, = 8;                                                                   SRC  485
MINDEN, = 2;                                                                    SRC  486
MAKELAR,= PROC(X),                                                              SRC  487
            BEGIN(J,I,II,NEWSIZE,NEWHT,IND),                                    SRC  488
               NEWSIZE = 2* HTSIZE,(X),                                         SRC  489
               NEWHT = MAKVECTO(NEWSIZE),                                       SRC  490
               FOR J=(1,HTSIZE,(X))REPEAT DO                                    SRC  491
                  I=HASHTAB,(X)[J],                                             SRC  492
                  WHILE I NE NIL REPEAT DO                                      SRC  493
                     II=HD I, I=TL I,                                           SRC  494
                     IND = COMPRHS,(NEWSIZE,HASHCOD,(II)),                      SRC  495
                     NEWHT[IND]=II;NEWHT[IND]                                   SRC  496
                  END                                                           SRC  497
               END,                                                             SRC  498
               X[3][2] = NEWSIZE,                                               SRC  499
               X[3][4] = NEWHT,                                                 SRC  500
               RETURN()                                                         SRC  501
            END                                                                 SRC  502
         END;                                                                   SRC  503
MAKESML,= PROC(S),                                                              SRC  504
            BEGIN(J,NEWSIZ,NEWHT),                                              SRC  505
               NEWSIZ=HTSIZE,(S) /2,                                            SRC  506
               NEWHT=MAKVECTO(NEWSIZ),                                          SRC  507
               FOR J=(1,NEWSIZ)REPEAT                                           SRC  508
               NEWHT[J]=APPEND,(HASHTAB,(S)[J],HASHTAB,(S)[J+NEWSIZ]),          SRC  509
               S[3][4]=NEWHT,S[3][2]=NEWSIZ,                                    SRC  510
               RETURN()                                                         SRC  511
            END                                                                 SRC  512
         ;                                                                      SRC  513
CHECKFG, = PROC(X),                                                             SRC  514
            IF HTLOAD, (X)GT                                                    SRC  515
            (MAXDEN, * HTSIZE,(X))THEN                                          SRC  516
            MAKELAR,(X)                                                         SRC  517
         END;                                                                   SRC  518
CHECKFS, = PROC(X),                                                             SRC  519
            BEGIN(Y),                                                           SRC  520
               IF NELT,(X) EQ 0 THEN X[3] = VECTOR(0,0,0,0) ELSE DO             SRC  521
                  Y = HTSIZE,(X),                                               SRC  522
                  IF HTLOAD,(X) * MINDEN, LE Y AND Y GT SMLHSTA, THEN           SRC  523
                  MAKESML,(X)
```

```
            END, RETURN()                                                    SRC  524
         END                                                                 SRC  525
      END;                                                                   SRC  526
COMMENT * NEWSMS FINDS A SUITABLE HASHTABLE SIZE FOR A SET OF N ELEMS*       SRC  527
NEWSMS, =  PROC(N),                                                          SRC  528
            IF N LT 16 THEN 8                                                SRC  529
            ELSEIF N LT 32 THEN 16                                           SRC  530
            ELSEIF N LT 64 THEN 32                                           SRC  531
            ELSEIF N LT 128 THEN 64                                          SRC  532
            ELSEIF N LT 256 THEN 128                                         SRC  533
            ELSE 256                                                         SRC  534
         END;                                                                SRC  535
                                                                             SRC  536
                                                                             SRC  537
COMMENT * SETL FUNCTIONS AND OPERATIONS  *                                   SRC  538
                                                                             SRC  539
ATOM,=     PROC(O),                                                          SRC  540
            IF NOT VECTQ(O) THEN TRUE ELSE TYPE,(O) LT SET.                  SRC  541
         END;                                                                SRC  542
EQUALT, =  PROC(A,B),                                                        SRC  543
            BEGIN( $Y,$TUPA,$TUPB,I,J,N,M,$X,$YY,W,WW);                      SRC  544
               IF A EQ UNDEF, THEN RETURN(IF B EQ UNDEF, THEN TRUE           SRC  545
               ELSE FALSE),                                                  SRC  546
               IF TYPE,(A) NE TYPE,(B) THEN RETURN(FALSE),                   SRC  547
               IF STRQ(A) THEN RETURN(EQSTR( ,B)),                           SRC  548
               IF BLANKQ,(A) THEN RETURN(A[3] EQ B[3]);                      SRC  549
               IF ATOM,(A) THEN RETURN(A EQ  ),                              SRC  550
               IF HASHCOD,(A) NE HASHCOD,(B)THEN RETURN (NIL),               SRC  551
               IF NELTS,(A) NE NELTS,(B) THEN RETURN(FALSE),                 SRC  552
               IF A EQ B THEN RETURN (TRUE),                                 SRC  553
               W=TRUE,                                                       SRC  554
               IF TYPE,(A) EQ TUPL, THEN DO                                  SRC  555
                  TUPA = TUPLE, (A),TUPB = TUPLE, (B),                       SRC  556
                  WHILE W AND TUPA NE NIL REPEAT DO                          SRC  557
                     X=HD TUPA, TUPA=TL TUPA,                                SRC  558
                     Y=HD TUPB, TUPB=TL TUPB,                                SRC  559
                     IF NOT EQUALT,(X,Y) THE  W=NIL                          SRC  560
                  END, RETURN(W)                                             SRC  561
               END,                                                          SRC  562
               IF HTSIZE,(A) LT HTSIZE,(B) THEN DO                           SRC  563
                  I=A,A=B,B=I                                                SRC  564
               END,N=HTSIZE,(B),                                             SRC  565
               J = HASHTAB,(A), M = HASHTAB,(B),                             SRC  566
               FOR I=(1,HTSIZE,(A))REPEAT DO                                 SRC  567
                  TUPA = J [I],                                              SRC  568
                  TUPB=M[COMPRHS,(N,I-1)],                                   SRC  569
                  WHILE TUPA NE NIL REPEAT DO                                SRC  570
                     X=HD TUPA, TUPA=TL TUPA, YY=TUPB, WW=NIL,               SRC  571
                     X=MAKOUTU,(X),                                          SRC  572
                     WHILE YY NE NIL REPEAT DO                               SRC  573
                        Y=HD YY, YY=TL YY, Y=MAKOUTU,(Y),                    SRC  574
                        IF EQUALT, (X,Y) THEN DO                             SRC  575
                           WW=TRUE,YY=NIL                                    SRC  576
                        END                                                  SRC  577
                     END,                                                    SRC  578
                     IF NOT WW THEN DO                                       SRC  579
                        W=NIL,I=HTSIZE,(A),TUPA=NIL                          SRC  580
                     END                                                     SRC  581
                  END                                                        SRC  582
               END,                                                          SRC  583
```

```
                RETURN(W)                                                    SRC  584
              END                                                            SRC  585
            END;                                                             SRC  586
NEQUALT,=   PROC(A,B),                                                       SRC  587
              IF EQUALT,(A,B) THEN FALSE ELSE TRUE                           SRC  588
            END;                                                             SRC  589
COMMENT ≠ FUNCTION ELTF, : ELEMENT OF, ≠                                     SRC  590
ELTF, =     PROC(X,S),                                                       SRC  591
              BEGIN(),                                                       SRC  592
                IF X EQ UNDEF, THEN ERRORTY,(                                SRC  593
                ≠EL(X,S),X IS UNDEFINED,≠),                                  SRC  594
                IF SETQQ,(S) THEN RETURN ELTFSET,(X,S),                      SRC  595
                IF TUPQ,(S) THEN RETURN ELTUPL, (X,S),                       SRC  596
                IF STRQ(S) THEN RETURN ELTFCST, (X,S),                       SRC  597
                IF BSTRQ,(S) THEN RETURN ELBITSTR (X,S),                     SRC  598
                ERRORTY, (≠EL(X,S),S NOT A SET,TUPLE,BSTRING,                SRC  599
                  OR CSTRING≠)                                               SRC  600
                END                                                          SRC  601
            END;                                                             SRC  602
COMMENT     ≠AUXILIARY ROUTINES FOR ELTF, ≠                                  SRC  603
ELTUPL,=    PROC( X,S),                                                      SRC  604
              BEGIN($Y,$Z),                                                  SRC  605
                IF S EQ NULLTUP, THEN RETURN (FALSE),                        SRC  606
                Z = TUPLE, (S),                                              SRC  607
                WHILE Z NE NIL REPEAT DO                                     SRC  608
                  Y=HD Z, Z=TL Z,                                            SRC  609
                  IF EQUALT,(X,Y) THEN RETURN(TRUE)                          SRC  610
                END,                                                         SRC  611
                RETURN (FALSE)                                               SRC  612
              END                                                            SRC  613
            END ;                                                            SRC  614
ELTFCST,= PROC(X,S),                                                         SRC  615
              BEGIN(I),                                                      SRC  616
                IF NOT STRQ(X) OR SIZE X NE 1 THEN ERRORTY,(                 SRC  617
                ≠X EL S, S IS A STRING, BUT X IS NOT A LIKE STRING OF,       SRC  618
                LENGTH 1≠),                                                  SRC  619
                FOR I=(1,SIZE S) REPEAT IF X EQ SUBSTR(S,I,1) THEN           SRC  620
                RETURN(TRUE),                                                SRC  621
                RETURN(NIL)                                                  SRC  622
              END                                                            SRC  623
            END;                                                             SRC  624
ELTFSET,=   PROC(X,S),                                                       SRC  625
              BEGIN(HX,TX,$I,$II,IND,N),                                     SRC  626
                IF S EQ NULLSET, THEN RETURN(FALSE),                         SRC  627
                IND = COMPRHS,(HTSIZE,(S),HASHCOD,(X)),                      SRC  628
                IF SPCTUPQ,(X) THEN DO                                       SRC  629
                  HX = HEAD,(X), TX=TAIL,(X), I=HASHTAB,(S)[IND],            SRC  630
                  WHILE I NE NIL REPEAT DO                                   SRC  631
                    II=HD I, I=TL I, IF TUPQ,(II) THEN IF                    SRC  632
                    IDENTQ,(NELTS,(II),3) THEN IF EQUALT,(HX,                SRC  633
                    TUPLE,(II)[1]) THEN RETURN                               SRC  634
                    (ELTFSET,(TX,TUPLE,(II)[2]))                             SRC  635
                  END,                                                       SRC  636
                  RETURN(FALSE)                                              SRC  637
                END,                                                         SRC  638
                IF VECTQ(X) THEN N=NELTS,(X),                                SRC  639
                I = HASHTAB,(S)[IND],                                        SRC  640
                WHILE I NE NIL REPEAT DO                                     SRC  641
                  II=HD I, I=TL I,                                           SRC  642
                  IF TUPQ,(II) THEN (IF NELTS,(II) EQ N AND                  SRC  643
```

```
                EQUALT,(X,MAKOUTU,(II)) THEN RETURN(TRUE))                    SRC  644
                ELSEIF EQUALT,(X,II) THEN RETURN(TRUE)                        SRC  645
             END,                                                             SRC  646
             RETURN(FALSE)                                                    SRC  647
          END                                                                 SRC  648
       END;                                                                   SRC  649
COMMENT ≠ FUNCTIONS HEAD,,TAIL,,WITH,,LESS, ≠                                 SRC  650
HEAD, =    PROC (TT) ,                                                        SRC  651
          IF TT EQ NULT, THEN                                                 SRC  652
          UNDEF, ELSEIF NOT TUPQ,(TT) THEN                                    SRC  653
          ERRORTY, ( ≠ HEAD,(T), T IS DEFINED BUT                             SRC  654
          NOT A TUPLE,≠) ELSE HD TUPLE,(TT)                                   SRC  655
       END;                                                                   SRC  656
TAIL,=     PROC(TT),                                                          SRC  657
          BEGIN(NEWL,NEWT),                                                   SRC  658
             IF NOT TUPQ,(TT) THEN ERRORTY,(                                  SRC  659
             ≠TAIL, T, T NOT A TUPLE,≠),                                      SRC  660
             IF TT EQ NULT, THEN RETURN(COPY(UNDEF,)),                        SRC  661
             IF NELTS,(TT) LE 1 THEN RETURN(COPY(NULT,))                      SRC  662
             ELSE DO                                                          SRC  663
                NEWL=NELT,(TT)-1, NEWT=TL TUPLE,(TT),                         SRC  664
                RETURN (VECTOR(TUPL,,HASHCOD,( HD NEWT),                      SRC  665
                VECTOR (NEWL,NEWT)))                                          SRC  666
             END                                                              SRC  667
          END                                                                 SRC  668
       END;                                                                   SRC  669
                                                                              SRC  670
COMMENT ≠ TAILN, RETURNS THE TAIL, OF, A TUPLE, FROM N-TH ELEMENT ON ≠        SRC  671
                                                                              SRC  672
TAILN, =   PROC(TT,N),                                                        SRC  673
          BEGIN(NEWL,NEWT,I),                                                 SRC  674
             IF NOT TUPQ,(TT) THEN ERRORTY,(                                  SRC  675
             ≠TAILN,(T,N), T NOT A TUPLE,≠),                                  SRC  676
             IF N LT 1 THEN ERRORVA,(                                         SRC  677
             ≠TAILN,(T,N), N IS SMALLER THAN 1≠),                             SRC  678
             IF N GT NELTS,(TT) THEN RETURN(COPY(NULT,)),                     SRC  679
             IF N EQ 1 THEN RETURN(TT),                                       SRC  680
             NEWT=TUPLE,(TT),                                                 SRC  681
             FOR I=(1,N-1) REPEAT NEWT=TL NEWT,                               SRC  682
             NEWL=NELT,(TT)-N+1,                                              SRC  683
             RETURN(VECTOR(TUPL,,HASHCOD,(HD NEWT),                           SRC  684
             VECTOR(NEWL,NEWT)))                                              SRC  685
          END                                                                 SRC  686
       END;                                                                   SRC  687
COMMENT ≠ SUBSTR OPERATES (RIGHT-HAND SIDE ONLY) ON STRINGS AND               SRC  688
       TUPLES ,,,,,,,, ≠                                                      SRC  689
SUBSTR =  PROC(A,B,C),                                                        SRC  690
          BEGIN(T1,K,L,I),                                                    SRC  691
             IF NOT INTQ(B) OR NOT INTQ(C) THEN ERRORTY,(                     SRC  692
             ≠SUBSTR(A,B,C), B OR C NOT INTEGER≠),                            SRC  693
             IF C LT 0 THEN ERRORVA,(                                         SRC  694
             ≠SUBSTR(A,B,C), C IS NEGATIVE≠),                                 SRC  695
             IF B LT 1 THEN ERRORVA,(                                         SRC  696
             ≠SUBSTR(A,B,C), B NOT POSITIVE≠),                                SRC  697
             IF (B+C-1) GT NELT,(A) THEN RETURN(COPY(UNDEF,)),                SRC  698
             IF STRQ(A) THEN RETURN(SUB(A,B,C)),                              SRC  699
             IF NOT TUPQ,(A) THEN ERRORTY,(                                   SRC  700
             ≠SUBSTR(A,B,C), A NOT A STRING NOR A TUPLE,≠),                   SRC  701
             IF A EQ NULT, THEN RETURN(COPY(UNDEF,)),                         SRC  702
                IF C EQ 0 THEN RETURN(COPY(NULT,)),                           SRC  703
```

```
                T1=COPY(TUPLE,(A)),                                                    SRC  704
                IF C EQ 1 THEN DO                                                      SRC  705
                   T1=GETWD,(T1,B),                                                    SRC  706
                   RETURN(VECTOR(TUPL,,HASHCOD,(T1),VECTOR(1,T1:NIL)))                 SRC  707
                END,                                                                   SRC  708
                T1=SKIPWD,(T1,B-1),                                                    SRC  709
                K=T1,                                                                  SRC  710
                FOR I=(1,C-1) REPEAT K=TL K,                                           SRC  711
                TL K=NIL,                                                              SRC  712
                RETURN(VECTOR(TUPL,,HASHCOD,(HD T1),                                   SRC  713
                   VECTOR(C,T1)))                                                      SRC  714
             END                                                                       SRC  715
          END;                                                                         SRC  716
WITH,=     PROC(S,A),                                                                  SRC  717
             AUGMENT,(COPY(S),A)                                                       SRC  718
          END;                                                                         SRC  719
LESS, =    PROC(S,A),                                                                  SRC  720
             DIMINIS,(COPY(S),A)                                                       SRC  721
          END;                                                                         SRC  722
COMMENT ≠ AUXILIARY FUNCTIONS FOR WITH, AND LESS, ≠                                    SRC  723
AUGMENT, = PROC(S,A),                                                                  SRC  724
             IF SETQQ,(S) THEN AUGMNTS,(S,A) ELSE ERRORTY,                             SRC  725
             ( ≠ AUGMENT, (S,A), S IS NOT A SET. ≠ )                                   SRC  726
          END;                                                                         SRC  727
AUGMNTS, = PROC (S,A),                                                                 SRC  728
             IF A EQ UNDEF, THEN ERRORTY, ( ≠ AUGMENT,                                 SRC  729
             (S,A),A IS THE UNDEFINED ATOM ≠ ) ELSE                                    SRC  730
             IF TUPQ,(A) THEN AUGMNTP, (S,A)                                           SRC  731
             ELSE AUGMNTM, (S,A)                                                       SRC  732
          END;                                                                         SRC  733
AUGMNTM, = PROC(S,A),                                                                  SRC  734
             BEGIN(),                                                                  SRC  735
                RETURN(AUGMNT1, (S,MAKSETU,(A)))                                       SRC  736
             END                                                                       SRC  737
          END;                                                                         SRC  738
AUGMNTP, = PROC(S,TT),                                                                 SRC  739
             BEGIN(HT,TU,IND,OLDSZ,$X,$Y,W),                                           SRC  740
                IF NELT,(TT)LE 2 THEN                                                  SRC  741
                RETURN(AUGMNTM, (S,TT)),                                               SRC  742
                IF S EQ NULLSET, THEN DO                                               SRC  743
                   W=SETWITO,(COPY(TT)),                                               SRC  744
                   S[2]=W[2],                                                          SRC  745
                   S[3]=COPY(W[3]),                                                    SRC  746
                   RETURN(S)                                                           SRC  747
                END,                                                                   SRC  748
                HT = HEAD, (TT),TU = TAIL, (TT),                                       SRC  749
                IF S NE NULLSET, THEN DO                                               SRC  750
                   IND = COMPRHS, (HTSIZE,(S),                                         SRC  751
                   HASHCOD, (HT)), Y = HASHTAB, (S) [ IND ],                           SRC  752
                   W = TRUE,                                                           SRC  753
                   WHILE Y NE NIL AND W REPEAT DO                                      SRC  754
                      X=HD Y, Y=TL Y,                                                  SRC  755
                      IF SPCTUPQ,(X) AND                                               SRC  756
                      EQUALT,(X[3][2][1],HT) THEN DO                                   SRC  757
                         W=NIL,                                                        SRC  758
                         OLDSZ=NELT,(X[3][2][2]),                                      SRC  759
                         X[3][2][2]=AUGMNTP,(X[3][2][2],TU),                           SRC  760
                         IF OLDSZ NE NELT,(X[3][2][2]) THEN                            SRC  761
                         S[3][1] = S[3][1] + 1, RETURN(S)                              SRC  762
                      END                                                              SRC  763
```

```
                    END                                                                    SRC  764
                 END,                                                                      SRC  765
                 RETURN(AUGMNTM,(S,VECTOR(TUPL,,HASHCOD,(HT),                              SRC  766
                 VECTOR(3,LIST(HT,SETWITO,(TU),0)))))                                      SRC  767
              END                                                                          SRC  768
           END;                                                                            SRC  769
AUGMNT1, = PROC(S,A),                                                                      SRC  770
           BEGIN( HT,IND, AA,$I,$X),                                                       SRC  771
              IF S EQ NL, THEN DO                                                          SRC  772
                 HT = MAKVECTO(SMLHSTA,),                                                  SRC  773
                 HT[COMPRHS, (SMLHSTA,,HASHCOD,(A))]=LIST(A),                              SRC  774
                 S[2]=HASHSTW,(NULLSET,,A),                                                SRC  775
                 S[3]=VECTOR(1,SMLHSTA,,1,HT),                                             SRC  776
                 RETURN(S)                                                                 SRC  777
              END,                                                                         SRC  778
              IND = COMPRHS, (HTSIZE,(S), HASHCOD,(A)),                                    SRC  779
              I = HASHTAB, (S) [ IND ],                                                    SRC  780
              AA=MAKOUTU,(A),                                                              SRC  781
              WHILE I NE NIL REPEAT DO                                                     SRC  782
                 X=HD I, I=TL I,                                                           SRC  783
                 IF EQUALT,(MAKOUTU,(X),AA) THEN                                           SRC  784
                 RETURN(S)                                                                 SRC  785
              END,                                                                         SRC  786
              S[2] = HASHSTW,(S,A),                                                        SRC  787
              S[3][1]=S[3][1] + 1,                                                         SRC  788
              S[3][3]=S[3][3]+1,                                                           SRC  789
              S[3][4][IND]=A:S[3][4][IND],                                                 SRC  790
              CHECKFG,(S), RETURN(S)                                                       SRC  791
           END                                                                             SRC  792
        END;                                                                               SRC  793
COMMENT * FUNCTION GENERATE SET. WITH, ONLY 1 ELEMENT *                                    SRC  794
SETWITO, = PROC(A),                                                                        SRC  795
           BEGIN(I,HT,S),                                                                  SRC  796
              HT=MAKVECTO (SMLHSTA,),                                                      SRC  797
              I = COMPRHS,(SMLHSTA,,HASHCOD,(A)),                                          SRC  798
              IF SPCTUPQ,(A) THEN DO                                                       SRC  799
                 S= SETWITO, (TAIL,(A)),HT[I]=                                             SRC  800
                 LIST(VECTOR(TUPL,,HASHCOD,(A), VECTOR(3,VECTOR(HEAD,                      SRC  801
                   (A),S,0))))                                                             SRC  802
              END                                                                          SRC  803
              ELSE                                                                         SRC  804
              HT[I]=LIST(IF TUPQ,(A) THEN MAKSETU,(A)ELSE A),                              SRC  805
              RETURN(VECTOR (SET,,HASHSTW,(NULLSET,,A),                                    SRC  806
              VECTOR(1,SMLHSTA,,1,HT)))                                                    SRC  807
           END                                                                             SRC  808
        END;                                                                               SRC  809
TUPWITO, = PROC(A),                                                                        SRC  810
           VECTOR(TUPL,,HASHCOD,(A),                                                       SRC  811
           VECTOR(1,LIST(A)))                                                              SRC  812
        END;                                                                               SRC  813
COMMENT * MAKE A SET, FROM A LIST OF, ELEMENTS *                                           SRC  814
SETFRML,= PROC(L,N),                                                                       SRC  815
           BEGIN(M,X,S),                                                                   SRC  816
              S=COPY(NL,),                                                                 SRC  817
              IF NILQ(L) THEN RETURN(S),                                                   SRC  818
              M=SMLHSTA,,                                                                  SRC  819
              SMLHSTA,=NEWSMS,(N),                                                         SRC  820
              WHILE L REPEAT DO                                                            SRC  821
                 X=HD L, L=TL L,                                                           SRC  822
                 AUGMNTS,(S,X)                                                             SRC  823
```

```
                END,
                SMLHSTA,=M, RETURN(S)
            END
        END;
DIMINIS, = PROC (S,A),
            IF SETQQ,(S) THEN
            DIMINST, (S,A)ELSE ERRORTY,(
            ≠ DIMINIS,(S,A), OR LESS, (S,A) S IS NOT A SET.≠)
        END;
DIMINST, = PROC(S,A),
            BEGIN(I,IND ,SUBTUP,J),
                IF A EQ UNDEF, THEN ERRORTY,( ≠ DIMINIS, (S,A),
                A IS THE UNDEFINED ATOM.≠),
                IF S EQ NULLSET, THEN RETURN (S),
                IF TUPQ,(A) THEN RETURN(DIMINSP, (S,A)),
                RETURN(DIMINSM, (S,A))
            END
        END;
DIMINSM, = PROC(S,A),
            BEGIN(AA,IND,HE,BCK,$SUBTUP,$I),
                IND =COMPRHS, (HTSIZE,(S), HASHCOD,(A)),
                SUBTUP=HASHTAB, (S) [IND],
                AA=MAKOUTU,(A),
                HE = BCK = NIL:NIL,
                WHILE SUBTUP NE NIL REPEAT DO
                    I=HD SUBTUP, SUBTUP=TL SUBTUP,
                    IF EQUALT,(MAKOUTU,(I),AA) THEN DO
                        IF TL HE EQ NIL THEN HASHTAB,(S)[IND]=SUBTUP
                        ELSE HASHTAB,(S)[IND]=APPEND,(TL HE,SUBTUP),
                        S [2] = HASHSTL,(S,A),
                        S [3] [1] = S [3] [1] +1 ,
                        S [3] [3] = S [3] [3] +1 ,
                        CHECKFS,(S),RETURN(S),SUBTUP=NIL
                    END,
                    BCK=ADDON(BCK,I)
                END,
                IF NELT,(S) EQ 0 THEN S[3]=NULLSET,[3],
                RETURN (S)
            END
        END;
DIMINSP, = PROC(S,TU),
            BEGIN(HT,TT,IND,OLDSZ,$X,$Y),
                IF NELT,(TU)LE 2 THEN RETURN(DIMINSM, (S,TU)),
                HT = HEAD,(TU),TT = TAIL,(TU),
                IND = COMPRHS, (HTSIZE,(S),HASHCOD,(HT)),
                Y=HASHTAB, (S) [IND],
                WHILE Y NE NIL REPEAT DO
                    X=HD Y, Y=TL Y,
                    IF SPCTUPQ,(X)
                    AND EQUALT,(HEAD,(MAKOUTU,(X)),HT) THEN DO
                        OLDSZ=NELTS,(TUPLE,(X)[2]),X[3][2][2]=
                        DIMINSP, (TUPLE,(X)[2],TT),
                        IF NELTS,(TUPLE,(X)[2]) NE OLDSZ THEN DO
                            IF IDENTQ,(NELTS,(TUPLE,(X)[2]),0) THEN
                            DIMINSM, (S,X) ELSE S[3][1]=S [3][1]+1
                            ,RETURN(S)
                        END
                    END
                END,
                RETURN(S)
```

```
            END                                                                  SRC  884
          END;                                                                   SRC  885
COMMENT ≠ FUNCTION LESF, AND AUXILIARY ROUTINES ≠                                SRC  886
LESF, =    PROC(FF,X),                                                           SRC  887
             IF NOT SETQQ,(FF) THEN                                              SRC  888
             ERRORTY, (≠LESSF(F,X),F NOT A SET,≠)                                SRC  889
             ELSEIF X EQ UNDEF, THEN ERRORTY,(                                   SRC  890
             ≠LESF,(F,X), X UNDEFINED≠)                                          SRC  891
             ELSE LESSFOK,(COPY(FF),X)                                           SRC  892
          END;                                                                   SRC  893
DIMF, =    PROC(FF,X),                                                           SRC  894
             IF NOT SETQQ,(FF) THEN ERRORTY,(                                    SRC  895
             ≠DIMF,(F,X) , F NOT A SET,≠)                                        SRC  896
             ELSEIF X EQ UNDEF, THEN ERRORTY,(≠DIMF,(F,X),X UNDEFINED≠)          SRC  897
             ELSE LESSFOK,(FF,X)                                                 SRC  898
          END;                                                                   SRC  899
LESSFOK,=  PROC(G,X),                                                            SRC  900
             BEGIN(IND,$Z,$Y,$Z1),                                               SRC  901
                IF G EQ NULLSET, THEN RETURN(G),                                 SRC  902
                IND=COMPRHS,(HTSIZE,(G),HASHCD,(X)),                             SRC  903
                Z=HASHTAB,(G)[IND], Z1=NIL,                                      SRC  904
                WHILE Z NE NIL REPEAT DO                                         SRC  905
                   Y=HD Z, Z=TL Z,                                               SRC  906
                   IF TUPQ,(Y)AND NELTS,(Y) GE 2 AND                             SRC  907
                   EQUALT,(TUPLE,(Y)[1],X) THEN DO                               SRC  908
                      G[3][1]=G[3][1]-IF NELTS,(Y) EQ 2 THEN 1                   SRC  909
                      ELSE NELTS,(TUPLE,(Y)[2]),                                 SRC  910
                      G[2] = HASHSTL,(G,Y),                                      SRC  911
                      G[3][3] = G[3][3] - 1                                      SRC  912
                   END                                                           SRC  913
                   ELSE Z1=APPEND,(Z1,Y)                                         SRC  914
                END,                                                             SRC  915
                G[3][4][IND]=Z1,                                                 SRC  916
                IF NELTS,(G) EQ 0 THEN G[3]=VECTOR(0,0,0,0)                      SRC  917
                ELSE CHECKFS,(G),                                                SRC  918
                RETURN(G)                                                        SRC  919
             END                                                                 SRC  920
          END;                                                                   SRC  921
                                                                                 SRC  922
COMMENT ≠ LESFN, AND DIMFN, ARE SIMILAR TO LESF, AND DIMF,,BUT WORK              SRC  923
          FOR LISTS OF, ARGUMENTS ,,,,,, ≠                                       SRC  924
                                                                                 SRC  925
LESFN, =   PROC(F,X),                                                            SRC  926
             IF NOT SETQQ,(F) THEN ERRORTY,(                                     SRC  927
             ≠LESFN,(F,X),F NOT A SET,≠)                                         SRC  928
             ELSEIF NOT PAIRQ(X) THEN ERRORTY,(                                  SRC  929
             ≠LESFN,(F,X), X NOT A LIST≠)                                        SRC  930
             ELSE LESFNOK,(COPY(F),X)                                            SRC  931
          END;                                                                   SRC  932
                                                                                 SRC  933
DIMFN, =   PROC(F,X),                                                            SRC  934
             IF NOT SETQQ,(F) THEN ERRORTY,(                                     SRC  935
             ≠DIMFN,(F,X), F NOT A SET,≠)                                        SRC  936
             ELSEIF NOT PAIRQ(X) THEN ERRORTY,(                                  SRC  937
             ≠DIMFN,(F,X), X NOT A LIST≠)                                        SRC  938
             ELSE LESFNOK,(F,X)                                                  SRC  939
          END;                                                                   SRC  940
                                                                                 SRC  941
                                                                                 SRC  942
LESFNOK, = PROC(F,X),                                                            SRC  943
```

```
            BEGIN(P,Y,SUBS,SB,YP,S),                                          FEB73  1
               S=NIL,                                                         SRC  945
               SUBS=X,                                                        SRC  946
               IF F EQ NL, THEN RETURN(F),                                    SRC  947
               P=COPY(NILVECT,), Y=NEXTELT,(F,P),                             SRC  949
               WHILE Y NE UNDEF, REPEAT DO                                    SRC  950
               YP=Y,  SB=SUBS,                                                FEB73  2
               WHILE SB AND TUPQ,(YP) AND EQUALT,(HEAD,(YP),HD SB)            FEB73  3
                  REPEAT DO SB=TL SB, YP=TAILSPC,(YP)                         FEB73  4
                  END,                                                        FEB73  5
               IF NULL(SB) THEN S=Y;S,                                        FEB73  6
               Y=NEXTELT,(F,P)                                                SRC  964
               END,                                                           SRC  965
               WHILE S NE NIL REPEAT DO                                       SRC  966
                  DIMINIS,(F,HD S),                                           SRC  967
                  S=TL S                                                      SRC  968
               END,                                                           SRC  969
               RETURN(F)                                                      SRC  970
            END                                                               SRC  971
         END;                                                                 SRC  972
COMMENT ≠ TAILSPC, USUALLY RETURNS THE TAIL, OF, A TUPLE, T, BUT IF           SRC  973
THIS TAIL IS ITSELF A TUPLE OF 1 ELEMENT, THEN THIS ELEMENT IS                FEB73  7
RETURNED ,,, ≠                                                                FEB73  8
TAILSPC,= PROC(T),                                                            SRC  976
            BEGIN(TT),                                                        FEB73  9
               TT=TAIL,(T),                                                   SRC  978
               IF TUPQ,(TT) AND NELTS,(TT) EQ 1 THEN TT=HEAD,(TT),            FEB73 10
               RETURN(TT)                                                     SRC  981
            END                                                               SRC  982
         END;                                                                 SRC  983
                                                                              SRC  984
COMMENT ≠ A FUNCTION TO TEST WHETHER X IS A TUPLE, WITH, AT LEAST             SRC  985
         2 ELEMENTS ≠                                                         SRC  986
                                                                              SRC  987
PAIRTUP,= PROC(X),                                                            SRC  988
            IF TUPQ,(X) AND NELTS,(X) GT 1 THEN TRUE ELSE FALSE               SRC  989
         END;                                                                 SRC  990
                                                                              SRC  991
COMMENT ≠FUNCTION ARB, RETURNS AN ARBITRARY ELEM, OF A SET OR A TUPLE≠        SRC  992
                                                                              SRC  993
ARB, =     PROC(S),                                                           SRC  994
             IF SETQQ,(S) THEN ARBSET,(S)ELSEIF TUPQ,(S) THEN HEAD,(S)        SRC  995
               ELSE ERRORTY,(≠ARB,(S), S NOT A SET OR A TUPLE≠)               SRC  996
         END;                                                                 SRC  997
ARBSET, =  PROC(S),                                                           SRC  998
             BEGIN(X,Y),                                                      SRC  999
                IF S EQ NULLSET, THEN RETURN (UNDEF,),                        SRC 1000
                X=ARBSIM,(S), IF NOT TUPQ,(X) THEN RETURN (X),                SRC 1001
                IF NELTS,(X) LE 2 THEN RETURN(MAKOUTU,(X)),                   SRC 1002
                Y= ARBSET, (TUPLE,(X) [2]),                                   SRC 1003
                X[3][2] = TUPLE,(X)[1] ; TUPLE,(Y),                           SRC 1004
                X[3][1] = NELT,(Y) +1, RETURN(X)                              SRC 1005
             END                                                              SRC 1006
         END;                                                                 SRC 1007
ARBSIM, =  PROC(S),                                                           SRC 1008
             BEGIN(X,I),                                                      SRC 1009
                X = HASHTAB,(S),FOR I=(1,HTSIZE,(S)) REPEAT                   SRC 1010
                IF X[I] NE NIL THEN RETURN(COPY(HD(X[I]))),                   SRC 1011
                ERRORIM,( ≠ A SET NE NL HAS NO ENTRIES IN ITS HASHTBL         SRC 1012
               ≠) END                                                         SRC 1013
         END;
```

```
                                                                        SRC 1014
COMMENT ≠ AUGMENT SET S WITH A COPY OF EACH OF THE MEMBERS OF SET P ≠   SRC 1015
AUGUNIN, = PROC(S,P),                                                   SRC 1016
             IF NOT SETQQ,(S)                                           SRC 1017
             OR NOT SETQQ,(P) THEN ERRORTY,                             SRC 1018
             ( ≠ AUGMENT, UNION,(S,P),EITHER S OR P IS NOT A SET,≠),    SRC 1019
             AUGUNIK,(S,P)                                              SRC 1020
          END;                                                          SRC 1021
AUGUNIK, = PROC(S,P),                                                   SRC 1022
             BEGIN (FLAG,J,QS,SS,I, $X,IND,W,Y,$Z,FLAG1,OLDSZ),         SEPT7335
                FLAG=NIL,QS=S,                                          SRC 1024
                IF NELTS,(S) GE NELTS,(P) THEN SS=P ELSE DO             SRC 1025
                   SS=S,S=P,FLAG=TRUE                                   SRC 1026
                END, J=HASHTAB,(SS),                                    SRC 1027
                FOR I = (1,HTSIZE,(SS))REPEAT DO                        SRC 1028
                   Z=J[I],                                              SRC 1029
                   FLAG1=TRUE,                                          SRC 1030
                   WHILE Z NE NIL REPEAT DO                             SRC 1031
                      X=HD Z, Z=TL Z, IF NOT TUPQ,(X) OR NELT,(X)       SRC 1032
                      NE 3 THEN AUGMNT1, (S,X) ELSE DO                  SRC 1033
                         IND = COMPRHS,(HTSIZE,(S),HASHCOD,(X)),        SRC 1034
                         W = HASHTAB, (S) [ IND ],                      SRC 1035
                         WHILE W NE NIL REPEAT DO                       SRC 1036
                            Y=HD W, W=TL W,                             SRC 1037
                            IF SPCTUPQ,(Y)                              SRC 1038
                            AND EQUALT, (TUPLE,(Y) [1],TUPLE,(X)[1])    SRC 1039
                            THEN DO                                     SRC 1040
                               OLDSZ=NELTS,(TUPLE,(Y)[2]),              SEPT7336
                               AUGUNIK, (TUPLE,(Y)[2],TUPLE,(X)[2]),    SRC 1041
                               FLAG1=W=NIL                              SRC 1042
                               , IF NELTS,(TUPLE,(Y)[2]) NE OLDSZ THEN  SEPT7337
                                 NELTS,(S)=NELTS,(S)+1                  SEPT7338
                            END                                         SRC 1043
                         END,                                           NOV73  1
                         IF FLAG1 THEN DO                               NOV73  2
                            AUGMNT1,(S,X),                              NOV73  3
***   X MAY BE SEVERAL ELEMENTS FOR EXAMPLE SEVERAL N TUPLES STARTING   NOV73  4
***       WITH THE SAME ELEMENT     -                                   NOV73  5
***       WHEN ADDING THEM TO A SET WHICH HAS NO NTUPLE STARTING WITH   NOV73  6
***       THAT ELEMENT THEN THE NUMBER OF ELEMENTS OF THE SET MUST BE   NOV73  7
***       ADJUSTED                                                      NOV73  8
                            NELTS,(S)=NELTS,(S)+NELTS,(TUPLE,(X)[2])-1  NOV73  9
                            END                                         NOV73 10
                         END                                            SRC 1045
                      END                                               SRC 1046
                END,                                                    SRC 1047
                IF FLAG THEN DO                                         SRC 1048
                   QS[1]=S[1],QS[2]=S[2],QS[3]=S[3]                     SRC 1049
                END,                                                    SRC 1050
                RETURN(S)                                               SRC 1051
             END                                                        SRC 1052
          END;                                                          SRC 1053
                                                                        SRC 1054
                                                                        SRC 1055
COMMENT ≠ REDEFINITION OF, PLUS,MINUS,TIMES, ETC,, ≠                    SRC 1056
                                                                        SRC 1057
PLS, =     PROC(X,Y),                                                   SRC 1058
             IF TYPE,(X) NE TYPE,(Y) THEN ERRORTY,(                     SRC 1059
             ≠A+B, TYPES OF, A AND B NOT IDENTICAL,≠)                   SRC 1060
             ELSEIF INTQ(X) THEN PLUSVD,(X,Y)                           SRC 1061
```

```
            ELSEIF STRQ(X) THEN STR,(X,Y)                                    SRC 1062
            ELSEIF SETQQ,(X) THEN UNION,(X,Y)                                SRC 1063
            ELSEIF TUPQ,(X) THEN TPLUS,(X,Y)                                 SRC 1064
            ELSEIF BSTRQ,(X) THEN BPLUS(X,Y)                                 SRC 1065
            ELSE ERRORTY,(≠A*B, A NOT A SET,TUPLE,BSTR,CSTR,OR INT≠)         SRC 1066
         END;                                                                SRC 1067
                                                                             SRC 1068
TPLUS,=   PROC(T1,T2),                                                       SEPT7161
            MAKTUP,(NELTS,(T1)+NELTS,(T2),APPEND(TUPLE,(T1),                 SEPT7162
            TUPLE,(T2)))                                                     SRC 1071
         END;                                                                SRC 1072
UNION,=   PROC(A,B),                                                         SRC 1073
            IF NELTS,(A) GE NELTS,(B) THEN AUGUNIK,(COPY(A),COPY(B))         SRC 1074
            ELSE AUGUNIK,(COPY(B),COPY(A))                                   SRC 1075
         END;                                                                SRC 1076
MNS,=     PROC(X,Y),                                                         SRC 1077
            IF TYPE,(X) NE TYPE,(Y) THEN ERRORTY,(                           SRC 1078
            ≠A-B, TYPES OF, A AND B NOT IDENTICAL≠)                          SRC 1079
            ELSEIF INTQ(X) THEN MINSVD,(X,Y)                                 SRC 1080
            ELSEIF SETQQ,(X) THEN SETMNS,(X,Y)                               SRC 1081
            ELSE ERRORTY,(                                                   SRC 1082
            ≠A-B, A NOT AN INTEGER NOR A SET,≠)                              SRC 1083
         END;                                                                SRC 1084
SETMNS,=  PROC(X,Y),                                                         SRC 1085
            BEGIN(S,P,NEXT),                                                 SRC 1086
               S=COPY(X), P=COPY(NILVECT,), NEXT=NEXTELT,(Y,P),              SRC 1087
               WHILE NEXT NE UNDEF, REPEAT DO                                SRC 1088
                  DIMINST,(S,NEXT),                                          SRC 1089
                  NEXT=NEXTELT,(Y,P)                                         SRC 1090
               END,                                                          SRC 1091
               RETURN(S)                                                     SRC 1092
            END                                                              SRC 1093
         END;                                                                SRC 1094
TMS,=     PROC(A,B),                                                         SRC 1095
            IF INTQ(A) AND INTQ(B) THEN TIMESVD,(A,B)                        SRC 1096
            ELSEIF INTQ(A) AND STRQ(B) THEN REPS,(A,B)                       SRC 1097
            ELSEIF SETQQ,(A) AND SETQQ,(B) THEN INTRSCT,(A,B)                SRC 1098
            ELSE ERRORTY,(≠A*B≠)                                             SRC 1099
         END;                                                                SRC 1100
REPS,=    PROC(A,B),                                                         SRC 1101
            BEGIN(R,I),                                                      SRC 1102
               IF A LT 0 THEN ERRORVA,(                                      SRC 1103
               ≠A*B, B IS A STRING AND A IS A NEGATIVE INTEGER≠),            SRC 1104
               R=≠≠,                                                         SRC 1105
               FOR I=(1,A) REPEAT R=STR,(R,B),                               SRC 1106
               RETURN(R)                                                     SRC 1107
            END                                                              SRC 1108
         END;                                                                SRC 1109
INTRSCT,= PROC(A,B),                                                         SRC 1110
            BEGIN(SM,LG,P,X,L,I),                                            SRC 1111
              IF NELTS,(A) GE NELTS,(B) THEN DO                              SRC 1112
                  SM=B, LG=A                                                 SRC 1113
               END                                                           SRC 1114
               ELSE DO                                                       SRC 1115
                  SM=A, LG=B                                                 SRC 1116
               END,                                                          SRC 1117
               P=COPY(NILVECT,),X=NEXTELT,(SM,P),L=NIL, I=0,                 SRC 1118
               WHILE X NE UNDEF, REPEAT DO                                   SRC 1119
                  IF ELTFSET,(X,LG) THEN DO L=X:L, I=I+1 END,                SRC 1120
                  X=NEXTELT,(SM,P)                                           SRC 1121
               END,
```

```
                RETURN(SETFRML,(L,I))                                    SRC 1122
            END                                                          SRC 1123
        END;                                                             SRC 1124
SLSH, =  PROC(X,Y),                                                      SRC 1125
            IF INTQ(X) AND INTQ(Y) THEN                                  SRC 1126
            SLHSVD,(X,Y)                                                 SRC 1127
            ELSE  ERRORTY,(≠A/B, A OR B NOT INTEGER≠)                    SRC 1128
        END;                                                             SRC 1129
DSLSH,=  PROC(A,B),                                                      SRC 1130
            IF INTQ(A) AND INTQ(B) THEN REMAIND,(A,B)                    SRC 1131
            ELSEIF SETQQ,(A) AND SETQQ,(B) THEN SYMDIF,(A,B)             SRC 1132
            ELSE ERRORTY,(≠A//B, A OR B NOT INT OR SET,≠)                SRC 1133
        END;                                                             SRC 1134
SYMDIF,= PROC(A,B),                                                      SRC 1135
            SETMNS,(UNION,(A,B),INTRSCT,(A,B))                           SRC 1136
        END;                                                             SRC 1137
                                                                         SRC 1138
COMMENT ≠ SET, INCLUSION ≠                                               SRC 1139
                                                                         SRC 1140
INCS, =  PROC(X,Y),                                                      SRC 1141
            BEGIN(Z,P),                                                  SRC 1142
               IF NOT(SETQQ,(X)) OR NOT(SETQQ,(Y)) THEN ERRORTY,(        SRC 1143
               ≠X INCS, Y, Y OR X NOT A SET,≠),                          SRC 1144
               IF EQUALT,(X,Y) THEN RETURN(TRUE),                        SRC 1145
               P=COPY(NILVECT,), Z=NEXTELT,(Y,P),                        SRC 1146
               WHILE Z NE UNDEF, REPEAT DO                               SRC 1147
                  IF NOT(ELTFSET,(Z,X)) THEN RETURN(FALSE)               SRC 1148
                  , Z=NEXTELT,(Y,P)                                      OCT72  1
               END,                                                      SRC 1149
               RETURN(TRUE)                                              SRC 1150
            END                                                          SRC 1151
        END;                                                             SRC 1152
                                                                         SRC 1153
                                                                         SRC 1154
COMMENT ≠ GENERATE NEW BLANK,, ATOM, ≠                                   SRC 1155
                                                                         SRC 1156
NEWAT,=  PROC(),                                                         SRC 1157
            BEGIN(),                                                     SRC 1158
               NEWATNR,=NEWATNR,+1,                                      SRC 1159
               RETURN(VECTOR(BLANK,,,54321+NEWATNR,,NEWATNR,))           SRC 1160
            END                                                          SRC 1161
        END;                                                             SRC 1162
                                                                         SRC 1163
                                                                         SRC 1164
COMMENT ≠ SOME OTHER ARITHMETIC FUNCTIONS ,,,,,, ≠                       SRC 1165
                                                                         SRC 1166
MAX,=    PROC(A,B),                                                      SRC 1167
            IF INTQ(A) AND INTQ(B) THEN                                  SRC 1168
              (IF A GE B THEN A ELSE B)                                  SRC 1169
            ELSE ERRORTY,(≠MAX,(A,B), A OR B NOT INTEGER≠)               SRC 1170
        END;                                                             SRC 1171
MIN,=    PROC(A,B),                                                      SRC 1172
            IF INTQ(A) AND INTQ(B) THEN                                  SRC 1173
              (IF A LE B THEN A ELSE B)                                  SRC 1174
            ELSE ERRORTY,(≠MIN,(A,B), A OR B NOT INTEGER≠)               SRC 1175
        END;                                                             SRC 1176
ABS,=    PROC(A),                                                        SRC 1177
            IF INTQ(A) THEN (IF A GE 0 THEN A ELSE -A)                   SRC 1178
            ELSE ERRORTY,(≠ABS,(A), A NOT INTEGER≠)                      SRC 1179
        END;                                                             SRC 1180
```

```
COMMENT ≠ GENERATING FUNCTIONS FOR SETS AND TUPLES .....≠

GENSET, =  PROC(),
             BEGIN(S,I,N,M),
                S=COPY(NL,),N=NUMARGS(),M=SMLHSTA,,
                SMLHSTA,=NEWSMS,(N),
                FOR I=(1,N)REPEAT AUGMENT,(S,ARGUMENT(I)),
                SMLHSTA,=M,
                RETURN(S)
             END
          END;
GENTUP,=   PROC(),
             BEGIN(L,M,I),
                L=NIL,M=NUMARGS(),
                FOR I=(1,M)REPEAT L=ARGUMENT(M-I+1)!L,RETURN(MAKTUP,
                (M,L))
             END
          END;

COMMENT ≠ DECIMAL AND OCTAL CONVERSION ......≠

DEC,=      PROC(O),
             IF STRQ(O) THEN IFROMS,(O,10)
             ELSEIF INTQ(O) THEN SFROMI,(O,10)
             ELSE ERRORTY,(≠DEC,(O), O NOT A STRING OR AN INTEGER≠)
          END;
OCT,=      PROC(O),
             IF STRQ(O) THEN IFROMS,(O,8)
             ELSEIF INTQ(O) THEN SFROMI,(O,8)
             ELSE ERRORTY,(≠OCT,(O), O NOT A STRING OR AN INTEGER≠)
          END;


COMMENT ≠ THE TWO POWERSET FUNCTIONS .........≠

POW, =     PROC(S),
             BEGIN(X,Y),
                IF NOT SETQQ,(S) THEN ERRORTY,(
                ≠POW,(S), S NOT A SET,≠),
                IF S EQ NL, THEN RETURN(SETWITO,(COPY(NL,))),
                Y=ARB,(S),DIMINIS,(S,Y),X=POW,(S),
                S=AUGMENT,(S,Y),
                RETURN(AUGUNIK, (X,EACHWIT,(X,Y)))
             END
          END;
NPOW, =    PROC(N,S),
             BEGIN(Y,XA,XB),
                IF NOT(SETQQ,(S)AND INTQ(N)) THEN ERRORTY,(
                ≠NPOW,(N,S), N NOT INT OR S NOT SET,≠),
                IF N EQ 0 THEN RETURN(SETWITO,(COPY(NL,))),
                IF N GT NELTS,(S) THEN RETURN(COPY(NL,)),
                IF IDENTQ,(N,NELTS,(S)) THEN RETURN(SETWITO,(COPY(S))),
                Y=ARB,(S), DIMINIS,(S,Y),XA=NPOW,(N-1,S),XB=NPOW,(N,S),
                AUGMENT,(S,Y),
                RETURN(AUGUNIK, (XB,EACHWIT,(XA,Y)))
             END
          END;
```

```
COMMENT ≠ AUXILIARY FUNCTIONS FOR POW, AND NPOW, ..... ≠                    SRC 1241
                                                                            SRC 1242
COMMENT ≠ EACHWIT,(S,X) ASSUMES THAT S IS A SET, OF,,SAY, N SETS SI,        SRC 1243
          IT RETURNS A SET, CONSISTING OF, N SETS WHICH CONSIST OF,         SRC 1244
          THE SETS SI EACH AUGMENTED WITH, THE ELEMENT X ..... ≠            SRC 1245
                                                                            SRC 1246
EACHWIT,= PROC(S,X),                                                        SRC 1247
             BEGIN(P,Y,R),                                                  SRC 1248
                R=COPY(NULLSET,), P=COPY(NILVECT,), Y=NEXTELT,(S,P),        SRC 1249
                WHILE Y NE UNDEF, REPEAT DO                                 SRC 1250
                   AUGMENT,(R,WITH,(Y,X)), Y=NEXTELT,(S,P)                  SRC 1251
                END,                                                        SRC 1252
                RETURN(R)                                                   SRC 1253
             END                                                            SRC 1254
          END;                                                              SRC 1255
                                                                            SRC 1256
                                                                            SRC 1257
COMMENT ≠ THE FUNCTION RANDOM, RETURNS A PSEUDO-RANDOM, NUMBER ≠            SRC 1258
                                                                            SRC 1259
RANDOMR,=1; COMMENT ≠ INITIAL VALUE OF, THE SEED ≠                          SRC 1260
                                                                            SRC 1261
RANDOM, = PROC(),                                                           JAN74  1
             BEGIN(A,B,K),                                                  JAN74  2
                IF NUMARGS() EQ 0 THEN K=131071 ELSE K=ARGUMENT(1),         JAN74  3
                IF NOT INTQ(K) THEN ERRORTY,(                               SRC 1265
                ≠RANDOM,(K), K NOT INT≠),                                   SRC 1266
  START,                                                                    SRC 1267
                B=SHIFT(RANDOMR,,-6),                                       SRC 1268
                A=XOR(RANDOMR,,B),                                          SRC 1269
                B=SHIFT(LAND(A,63),11),                                     SRC 1270
                RANDOMR,=XOR(A,B),                                          SRC 1271
                A=RANDOMR,/(131071/K)+1,                                    SRC 1272
                IF A GT K THEN GO START,                                    SRC 1273
                RETURN(A)                                                   SRC 1274
             END                                                            SRC 1275
          END;                                                              SRC 1276
                                                                            SRC 1277
COMMENT ≠ FUNCTIONAL APPLICATION (P. 55) ..,,..... ≠                        SRC 1278
                                                                            SRC 1279
COMMENT  ≠ ,,, THE SET, OF, ALL IMAGES OF, X IN F ,,, ≠                     SRC 1280
SOF, =     PROC(F,X),                                                       SRC 1281
             DO                                                             SRC 1282
                IF NOT SETQQ,(F) THEN                                       SRC 1283
                ERRORTY, (≠ F SOF, X,F IS NOT A SET, ≠);                    SRC 1284
                IF X EQ UNDEF, THEN ERRORTY, ( ≠ F SOF, X,                  SRC 1285
                X IS THE UNDEFINED ATOM, ≠);                                SRC 1286
                SOFOK,(F,X)                                                 SRC 1287
             END                                                            SRC 1288
          END;                                                              SRC 1289
SOFOK, =   PROC(F,X),                                                       SRC 1290
             BEGIN(IND,S,$W,$Y,Y2),                                         SRC 1291
                IF F EQ NULLSET, THEN RETURN(COPY(NL,));                    SRC 1292
                IND = COMPRHS, (HTSIZE,(F),HASHCOD,(X));                    SRC 1293
                S=COPY(NULLSET,), W=HASHTAB,(F)[IND],                       SRC 1294
                WHILE W NE NIL REPEAT DO                                    SRC 1295
                   Y=HD W, W=TL W, IF TUPQ,(Y) AND                          SRC 1296
                   NELTS,(Y) GE 2 AND EQUALT,(TUPLE,(Y)[1],X)               SRC 1297
                   THEN DO                                                  SRC 1298
                      Y2= TUPLE,(Y) [2], IF NELT,(Y)EQ 2 THEN DO            SRC 1299
                         IF Y2 NE UNDEF, THEN S=                            SRC 1300
```

```
                        AUGMNTS,(S,Y2)                                          SRC 1301
                    END ELSE S=AUGUNIK, (S,Y2)                                  SRC 1302
                    END                                                         SRC 1303
                END,RETURN (S)                                                  SRC 1304
            END                                                                 SRC 1305
        END;                                                                    SRC 1306
SOFN,=   PROC(F,SBS),                                                           SRC 1307
            BEGIN($HX,IND,$S,$Q,$Y,$YP,$SUBS,$SB,$XI),                          SRC 1308
                IF NOT PAIRQ(SBS) THEN ERRORTY,(                                SRC 1309
                ≠F SOFN, X, X NOT A LIST≠),                                     SRC 1310
                IF NOT SETQQ,(F) THEN ERRORTY,(                                 SRC 1311
                ≠F SOFN, X, F NOT A SET,≠),                                     SRC 1312
                IF TL SBS EQ NIL THEN RETURN(SOF,(F,HD SBS)),                   SRC 1313
                SUBS=SBS,                                                       SRC 1314
                IF F EQ NL, THEN RETURN(COPY(NL,)),                             SRC 1315
                IF SUBS EQ NIL THEN ERRORTY,(≠SOFN,(F,X), X IS NIL,≠),          SRC 1316
                HX = HD SUBS,                                                   SRC 1317
                IF HX EQ UNDEF, THEN SOFNERR,(),                                SRC 1318
                IND = COMPRHS,(HTSIZE,(F), HASHCOD,(HX)),                       SRC 1319
                S=COPY(NL,), Q=HASHTAB,(F)[IND],                                SRC 1320
                WHILE Q NE NIL REPEAT DO                                        SRC 1321
                    Y=HD Q,                                                     SRC 1322
                    Q=TL Q,                                                     SRC 1323
                    IF TUPQ,(Y) AND NELTS,(Y) GE 2 AND                          SRC 1324
                    EQUALT,(TUPLE,(Y)[1],HX)                                    SRC 1325
                    THEN DO                                                     SRC 1326
                        IF NELTS,(Y) EQ 2 THEN DO                               SRC 1327
                            YP=COPY(TUPLE,(Y)[2]),                              SRC 1328
                            SB=TL SUBS,                                         SRC 1329
                            IF SB NE NIL AND NOT TUPQ,(YP) THEN YP=UNDEF,,      SRC 1330
                            WHILE SB NE NIL AND TUPQ,(YP) REPEAT DO             SRC 1331
                                XI=HD SB, SB=TL S ,                             SRC 1332
                                IF XI EQ UNDEF, THEN SOFNERR,(),                SRC 1333
                                IF NELT,(YP) GT 1 AND                           SRC 1334
                                EQUALT,(HEAD,(YP),XI) THEN                      SRC 1335
                                (YP=IF NELT,(YP) EQ 2 THEN HD TL TUPLE,(YP)     SRC 1336
                                ELSE TAIL,(YP)) ELSE YP=UNDEF.                  SRC 1337
                            END,                                                SRC 1338
                            IF YP NE UNDEF, THEN AUGMNTS,(S,YP)                 SRC 1339
                        END                                                     SRC 1340
                        ELSE DO                                                 SRC 1341
                            YP=TUPLE,(Y)[2], IF TL SUBS EQ NIL THEN             SRC 1342
                            AUGUNIK, (S,YP)ELSE  UGUNIK, (S,SOFN,(YP, TL        SRC 1343
                            SUBS))                                              SRC 1344
                            END                                                 SRC 1345
                        END                                                     SRC 1346
                    END,                                                        SRC 1347
                    RETURN(S)                                                   SRC 1348
                    END                                                         SRC 1349
        END;                                                                    SRC 1350
SOFNERR,=  PROC(),                                                              SRC 1351
            ERRORTY,(≠SOFN,(S,X), SOME ELEMENT OF, X IS UNDEFINED,≠)            SRC 1352
            END;                                                                SRC 1353
                                                                                SRC 1354
COMMENT  ≠ ,,, THE SINGLE IMAGE OF, A IN O ,,, ≠                                SRC 1355
OF, =      PROC(O,A),                                                           SRC 1356
            IF SETQQ,(O) THEN OFSET,(O,A)                                       SRC 1357
            ELSEIF TUPQ,(O) THEN OFTUPL,(O,A)                                   SRC 1358
            ELSEIF CODEQ(O) THEN O(A)                                           SRC 1359
            ELSEIF STRQ(O) THEN SUBSTR(O,                                       SRC 1360
```

```
            IF VAR A THEN A ELSE HD A,                                          SRC 1361
            IF VAR A THEN 1 ELSEIF TL A THEN TL A ELSE -1)                      SRC 1362
            ELSEIF BSTRQ,(0) THEN OFBSTR(0,A)                                   SRC 1363
            ELSE ERRORTY,(≠0 OF A, A NOT A S T, TUPLE, BSTRING,                 SRC 1364
            CSTRING, OR PROC≠)                                                  SRC 1365
            END;                                                                SRC 1366
OFSET, =    PROC(F,X),                                                          SRC 1367
            BEGIN(DEFINED,RESULT,Y,Y2,IND,W),                                   SRC 1368
               IF X EQ UNDEF, THEN ERRORVA,(                                    SRC 1369
               ≠OF,(F,X), X IS UNDEFINED≠),                                     SRC 1370
               IF F EQ NULLSET, THEN RETURN(UNDEF,),                            SRC 1371
               DEFINED=FALSE, IND=COMPRHS,(HTSIZE,(F),HASHCOD,(X)),             SRC 1372
               W=HASHTAB,(F)[IND],                                              SRC 1373
               WHILE W NE NIL REPEAT DO                                         SRC 1374
                  Y=HD W, W=TL W,                                               SRC 1375
                  IF TUPQ,(Y) AND NELTS,(Y) GE 2 AND                            SRC 1376
                  EQUALT,(TUPLE,(Y)[1],X) THEN DO                               SRC 1377
                     IF DEFINED THEN RETURN(UNDEF,),                            SRC 1378
                     DEFINED=TRUE,                                              SRC 1379
                     Y2=TUPLE,(Y)[2],                                           SRC 1380
                     IF IDENTQ,(NELTS,(Y),2) THEN RESULT=COPY(Y2) ELSE          SRC 1381
                     RESULT=                                                    SRC 1382
                     IF NELT,(Y2) EQ 1 THEN ARBSET,(Y2) ELSE UNDEF,             SRC 1383
                  END                                                           SRC 1384
               END,                                                             SRC 1385
               RETURN(IF DEFINED THEN RESULT ELSE UNDEF,)                       SRC 1386
            END                                                                 SRC 1387
         END;                                                                   SRC 1388
OFTUPL,=    PROC(TUP,B),                                                        SRC 1389
            BEGIN(K,J),                                                         SRC 1390
               K=NELTS,(TUP),                                                   SRC 1391
               IF INTQ(B) THEN DO                                               SRC 1392
                  IF B LT 1 THEN ERRORVA,(                                      SRC 1393
                  ≠TUPLE, OF, X, X LESS, THAN 1≠),                              SRC 1394
                  IF B GT K THEN RETURN(UNDEF,)                                 SRC 1395
               END,                                                             SRC 1396
               TUP=TUPLE,(TUP),                                                 SRC 1397
               IF VAR B THEN RETURN(COPY(GETWD,(TUP,B))),                       SRC 1398
               J = HD B, K=K+1-J,TUP=COPY(SKIPWD,(TUP,J-1)),                    SRC 1399
               IF TL B EQ NIL THEN RETURN(MAKTUP,(K,TUP)),                      SRC 1400
               J=TL B,                                                          SRC 1401
               RETURN(MAKTUP,(J,FIRSTWD,(TUP,J)))                               SRC 1402
            END                                                                 SRC 1403
         END;                                                                   SRC 1404
OFN, =      PROC(F,SBS),                                                        SRC 1405
            BEGIN(K),                                                           SRC 1406
               IF NOT PAIRQ(SBS) THEN ERRORTY,(                                 SRC 1407
               ≠F OFN, X,X NOT A LIST≠),                                        SRC 1408
               IF CODEQ(F) THEN RETURN(APPLY(F,SBS)),                           SRC 1409
               IF TL SBS EQ NIL THEN RETURN(OF,(F,HD SBS)),                     SRC 1410
               K=SOFN,(F,SBS),                                                  SRC 1411
               IF NELTS,(K) NE 1 THEN RETURN(UNDEF,),                           SRC 1412
               RETURN(ARBSET,(K))                                               SRC 1413
            END                                                                 SRC 1414
         END;                                                                   SRC 1415
                                                                                SRC 1416
COMMENT ≠ 3 AUXILIARY FUNCTIONS TO PROCESS LISTS .....≠                         SRC 1417
GETWD,=     PROC(L,N),                                                          SRC 1418
            BEGIN($X,I),                                                        SRC 1419
               X=UNDEF,, FOR I=(1,N) REPEAT DO                                  SRC 1420
```

```
                X=HD L, L=TL L                                                  SRC 1421
             END,                                                               SRC 1422
             RETURN(X)                                                          SRC 1423
          END                                                                   SRC 1424
       END;                                                                     SRC 1425
SKIPWD,=  PROC(L,N),                                                            SRC 1426
          BEGIN(X,I),                                                           SRC 1427
             FOR I=(1,N) REPEAT DO                                              SRC 1428
                X=HD L, L=TL L                                                  SRC 1429
             END,                                                               SRC 1430
             RETURN(L)                                                          SRC 1431
          END                                                                   SRC 1432
       END;                                                                     SRC 1433
FIRSTWD,= PROC(L,N),                                                            SRC 1434
          BEGIN(I,$X,$Q),                                                       SRC 1435
             Q=L, FOR I=(1,N-1) REPEAT DO                                       SRC 1436
                X=HD Q, Q=TL Q                                                  SRC 1437
             END,                                                               SRC 1438
             TL Q=NIL, RETURN(L)                                                SRC 1439
          END                                                                   SRC 1440
       END;                                                                     SRC 1441
COMMENT ≠ ITERATION OVER A COLLECTION (SEE PAGE 62 FOR DETAILS)                 SRC 1442
       ONE IMPORTANT DIFFERENCE WITH, NEWLETTER 49 IS THAT P                    SRC 1443
       SHOULD BE INITIALIZED TO  [NIL]  OR NILVECT,, BUT NOT                    SRC 1444
       TO NULT,,,,,,,,,, ≠                                                      SRC 1445
NEXTELT,= PROC(O,P),                                                            SRC 1446
          IF P[1] EQ UNDEF, THEN ERRORTY,(                                      SRC 1447
          ≠ITERATION OVER S, ITERATOR CALLED AFTER END OF, OBJECT               SRC 1448
          REACHED,≠)                                                            SRC 1449
          ELSEIF SETQQ,(O) THEN NXTLTS,(O,P)                                    SRC 1450
          ELSEIF TUPQ,(O) THEN NXTLTT,(O,P)                                     SRC 1451
          ELSEIF STRQ(O) THEN NXTLTC,(O,P)                                      SRC 1452
          ELSEIF BSTRQ,(O) THEN NXTLTB(O,P)                                     SRC 1453
          ELSEIF NOT VECTQ(O) THEN ERRORTY,(                                    SRC 1454
          ≠ITERATOR OVER S, S NOT A SET,, TUPLE, OR STRING,≠)                   SRC 1455
          ELSE ERRORTY,(                                                        SRC 1456
          ≠ITERATION OVER S, S NOT RECOGNIZABLE≠)                               SRC 1457
       END;                                                                     SRC 1458
NXTLTC,=  PROC(O,P),                                                            SRC 1459
          BEGIN(C),                                                             SRC 1460
             IF P[1] EQ NIL THEN P[1] = VECTOR(1,SIZE(O)),                      SRC 1461
             IF P[1][1] GT P[1][2] THEN DO                                      SRC 1462
                P[1]=UNDEF,, RETURN(UNDEF,)                                     SRC 1463
             END,                                                               SRC 1464
             C=SUBSTR(O,P[1][1],1), P[1][1]=P[1][1]+1,                          SRC 1465
             RETURN(C)                                                          SRC 1466
          END                                                                   SRC 1467
       END;                                                                     SRC 1468
NXTLTT2, = PROC(T,P),                                                           SRC 1469
          BEGIN(Q),                                                             SRC 1470
             IF P[1] EQ NIL THEN P[1]=VECTOR(1,TUPLE,(T)),                      SRC 1471
             IF P[1][1] GT NELTS,(T) THEN DO                                    SRC 1472
                P[1] = UNDEF,, RETURN(UNDEF,)                                   SRC 1473
             END,                                                               SRC 1474
             Q=HD P[1][2],                                                      SRC 1475
             P[1][2]=TL P[1][2], P[1][1]=P[1][1]+1,                             SRC 1476
             RETURN(Q)                                                          SRC 1477
          END                                                                   SRC 1478
       END;                                                                     SRC 1479
NXTLTT,=  PROC(T,P),                                                            SRC 1480
```

```
            BEGIN(X),                                                    SRC 1481
   GETELT,     X=NXTLTT2,(T,P),                                          SRC 1482
               IF P[1] EQ UNDEF, THEN RETURN(UNDEF,),                    SRC 1483
               IF X EQ UNDEF, THEN GO GETELT,                            SRC 1484
               RETURN(X)                                                 SRC 1485
            END                                                          SRC 1486
         END;                                                            SRC 1487
NXTLTS,=   PROC(S,P),                                                    SRC 1488
            BEGIN(ITEM),                                                 SRC 1489
               IF P[1] EQ NIL THEN P[1]=VECTOR(0,NIL,NIL,                SRC 1490
               MAKVECTO(1),NIL,0),                                       SRC 1491
               IF P[1][4][1] NE NIL THEN GO TUP,                         SRC 1492
  ENT,         IF P[1][2] EQ NIL THEN GO NXTLN,                          SRC 1493
  ENT1,        ITEM=HD P[1][2], P[1][2]=TL P[1][2],                      SRC 1494
               IF TUPQ,(ITEM)                                            SRC 1495
               THEN IF NELT,(ITEM) EQ 3 THEN GO TUP1, RETURN(MAKOUTU,(   SRC 1496
               ITEM)),                                                   SRC 1497
  TUP1,        P[1][3] = TUPLE,(ITEM)[1],                                SRC 1498
               P[1][6]=HASHCOD,(ITEM),                                   SRC 1499
               P[1][5]=TUPLE,(ITEM)[2],                                  SRC 1500
  TUP,         ITEM=NXTLTS,(P[1][5],P[1][4]),                            SRC 1501
               IF P[1][4][1] NE UNDEF, THEN RETURN(                      SRC 1502
              VECTOR(TUPL,,P[1][6],VECTOR(NELT,(ITEM)+1,P[1][3],         SRC 1503
               TUPLE,(ITEM)))),                                          SRC 1504
               P[1][4][1]=NIL, GO ENT,                                   SRC 1505
  NXTLN,       P[1][1] = P[1][1] + 1, IF P[1][1] GT HTSIZE,(S) THEN GO   SRC 1506
               TERM,                                                     SRC 1507
               P[1][2]=HASHTAB,(S)[P[1][1]],                             SRC 1508
               IF P[1][2] NE NIL THEN GO ENT1, GO NXTLN,                 SRC 1509
  TERM,        P[1] = UNDEF,, RETURN(UNDEF,)                             SRC 1510
            END                                                          SRC 1511
         END;                                                            SRC 1512
                                                                         SRC 1513
COMMENT ≠ THE UNION, OF, ALL SETS WHICH ARE THE IMAGES OF, ALL           SRC 1514
         THE ELEMENTS OF, S UNDER F ≠                                    SRC 1515
                                                                         SRC 1516
BOF, =     PROC(F,S),                                                    SRC 1517
            IF NOT SETQQ,(F) AND NOT CODEQ(F) THEN ERRORTY,(             SRC 1518
            ≠F BOF, S, F NOT A SET, NOR A PR C≠)                         SRC 1519
            ELSEIF CODEQ(F) THEN BOFCODE,(F,S)                           SRC 1520
            ELSEIF SETQQ,(S) THEN BOFSET,(F,S)                           SRC 1521
            ELSEIF TUPQ,(S) THEN BOFTUPL,(F,S)                           SR  1522
            ELSEIF STRQ(S) THEN BOFSTR(F,S)                              SRC 1523
            ELSEIF BSTRQ,(S) THEN BOFBSTR(F,S)                           SRC 1524
            ELSE ERRORTY,(                                               SRC 1525
            ≠BOF,(F,S), S NOT A SET,,TUPLE,,STRING,BSTRING≠)             SRC 1526
         END;                                                            SRC 1527
                                                                         SRC 1528
BOFSET, =  PROC(F,S),                                                    SRC 1529
            BEGIN(RESULT,NEXT,P),                                        SRC 1530
               IF S EQ NULLSET, THEN RETURN(COPY(NL,));                  SRC 1531
               P=COPY(NILVECT,),RESULT=COPY(NULLSET,),NEXT=NEXTELT,      SRC 1532
               (S,P),  WHILE NEXT NE UNDEF, REPEAT DO                    SRC 1533
                  AUGUNIK,(RESULT,SOFOK,(F,NEXT)),                       SRC 1534
                  NEXT=NEXTELT,(S,P)                                     SRC 1535
               END,                                                      SRC 1536
               RETURN(RESULT)                                            SRC 1537
            END                                                          SRC 1538
         END;                                                            SRC 1539
                                                                         SRC 1540
```

```
BOFTUPL, = PROC(F,T),                                                    SRC 1541
          BEGIN(RESULT,COUNT,TODO,I,X,Y,N),                              SRC 1542
             TODO=TUPLE,(T),RESULT=NIL,COU T=0,N=NELT,(T),               SRC 1543
             FOR I=(1,N) REPEAT DO                                       SRC 1544
                X=HD TODO, TODO=TL TODO,                                 SRC 1545
                IF NOT(X EQ UNDEF,) THEN DO                              SRC 1546
                Y=OFSET,(F,X),                                           SRC 1547
                IF Y EQ UNDEF, THEN ERRORVA,(                            SRC 1548
                ≠BOF(F,T), T IS A TUPLE AND F(T(I)) UNDEF FOR I≠),       SRC 1549
                APPEND,(RESULT,Y),                                       SRC 1550
                COUNT=COUNT+1                                            SRC 1551
                END                                                      SRC 1552
             END,                                                        SRC 1553
             IF RESULT=NIL THEN RETURN(UNDEF,),                          SRC 1554
             RETURN(VECTOR(TUPL,,HASHCOD,(HD RESULT),                    SRC 1555
             VECTOR(COUNT,RESULT)))                                      SRC 1556
          END                                                            SRC 1557
       END;                                                              SRC 1558
BOFCODE, = PROC(F,S),                                                    SRC 1559
          BEGIN(RESULT,NEXT,P),                                          SRC 1560
             IF S EQ NL, THEN RETURN(COPY(NL,)),                         SRC 1561
             IF NOT SETQQ,(S) THEN ERRORTY,(                             SRC 1562
             ≠BOFCODE,(F,S),S NOT A SET,≠),                              SRC 1563
             P=COPY(NILVECT,),RESULT=COPY( L,),NEXT=NEXTELT,(S,P),       SRC 1564
             WHILE NEXT NE UNDEF, REPEAT DO                              SRC 1565
                AUGMENT,(RESULT,F(NEXT)),                                SRC 1566
                NEXT=NEXTELT,(S,P)                                       SRC 1567
             END,                                                        SRC 1568
             RETURN(RESULT)                                              SRC 1569
          END                                                            SRC 1570
       END;                                                              SRC 1571
BOFN, =    PROC(F,SBS),                                                  SRC 1572
          IF NOT SETQQ,(F) AND NOT CODEQ(F) THEN ERRORTY,                SRC 1573
          (≠F BOFN, X, F NOT A SET, NOR A PROC≠)                         SRC 1574
          ELSEIF NOT PAIRQ(SBS) THEN ERRORTY,(                           SRC 1575
          ≠F BOFN, X,X NOT A LIST≠)                                      SRC 1576
          ELSEIF TL SBS EQ NIL THEN BOF,(F,HD SBS)                       SRC 1577
          ELSE BOFNOK,(F,SBS)                                            SRC 1578
       END;                                                              SRC 1579
BOFNOK, =  PROC(F,SBS),                                                  SRC 1580
          BEGIN(RESULT,NEXT),                                            SRC 1581
            RESULT=F,                                                    SRC 1582
             WHILE SBS NE NIL AND RESULT NE NL, REPEAT DO                SRC 1583
                NEXT=HD SBS,SBS=TL SBS,                                  SRC 1584
                RESULT=BOF,(RESULT,NEXT)                                 SRC 1585
             END,                                                        SRC 1586
             RETURN(RESULT)                                              SRC 1587
          END                                                            SRC 1588
       END;                                                              SRC 1589
COMMENT ≠ A PROCEDURE TO SELECT THE N=TH ELEMENT OF, A TUPLE, ≠          SRC 1590
SELTUPL, = PROC(T,N),                                                    SRC 1591
          BEGIN(),                                                       SRC 1592
             IF NOT TUPQ,(T) THEN ERRORTY,(                              SRC 1593
             ≠ T COMP N, T IS NOT A TUPLE,≠),                            SRC 1594
             IF NOT INTQ(N) THEN ERRORTY,(                               SRC 1595
             ≠ T COMP N, N IS NOT AN INTEGER≠),                          SRC 1596
             IF N LE 0 THEN ERRORVA,(                                    SRC 1597
             ≠ T COMP N, N NOT POSITIVE≠),                               SRC 1598
             IF N GT NELT,(T) THEN RETURN(UNDEF,),                       SRC 1599
             RETURN(COPY(GETWD,(TUPLE,(T),N)))                           SRC 1600
```

```
            END                                                                  SRC 1601
         END;                                                                    SRC 1602
COMMENT ≠ SINISTER CALLS FOR TUPLES AND FUNCTIONS ≠                              SRC 1603
SETSIN, =  PROC(A,B,C),                                                          SRC 1604
            IF TUPQ,(A) THEN SETTUP,(A,B,C)                                      SRC 1605
            ELSEIF SETQQ,(A) THEN SETFUN,(A,B,C) ELSE ERRORTY,(                  SRC 1606
            ≠LET A ,,, BE ,,,, A IS NOT A TUPLE, OR A SET,≠)                     SRC 1607
         END;                                                                    SRC 1608
                                                                                 SRC 1609
SETTUP, =  PROC(A,B,C),                                                          SRC 1610
            BEGIN(N,I,K),                                                        SRC 1611
               IF B LE 0 THEN ERRORVA,(                                          SRC 1612
               ≠LET T COMP I BE ,,,, I IS LESS, THAN 1≠),                        SRC 1613
               IF A EQ NULT, THEN DO                                             SRC 1614
                  IF B EQ 1 THEN DO                                              SRC 1615
                     A[2]=HASHCOD,(C),                                           SRC 1616
                     A[3][1]=1,                                                  SRC 1617
                     A[3][2]=COPY(C);NIL,                                        SRC 1618
                     RETURN(A)                                                   SRC 1619
                  END,                                                           SRC 1620
                  K=COPY(C);NIL,                                                 SRC 1621
                  FOR I=(1,B-1) REPEAT K=COPY(UNDEF,);K,                         SRC 1622
                  A[2]=HASHCOD,(UNDEF,),                                         SRC 1623
                  A[3][1]=B,                                                     SRC 1624
                  A[3][2]=K,                                                     SRC 1625
                  RETURN(A)                                                      SRC 1626
               END,                                                              SRC 1627
               IF B EQ 1 THEN DO                                                 SRC 1628
                  A[2]=HASHCOD,(C), HD(A[3][2])=C, RETURN(A)                     SRC 1629
               END,                                                              SRC 1630
               K=IF B LE NELTS,(A) THEN B-1 ELSE NELTS,(A)-1,                    SRC 1631
               N=TUPLE,(A),                                                      SRC 1632
               FOR I=(1,K) REPEAT N=TL N,                                        SRC 1633
               IF B LE NELTS,(A) THEN DO                                         SRC 1634
                  HD N=C, RETURN(A)                                              SRC 1635
               END,                                                              SRC 1636
               FOR I=(1,B-NELTS,(A)-1)REPEAT N=ADDON(N,COPY(UNDEF,)),            SRC 1637
               N=ADDON(N,COPY(C)),                                               SRC 1638
               A[3][1]=B,                                                        SRC 1639
               RETURN(A)                                                         SRC 1640
            END                                                                  SRC 1641
         END;                                                                    SRC 1642
                                                                                 SRC 1643
SETFUN, =  PROC(A,B,C),                                                          SRC 1644
            BEGIN(L),                                                            SRC 1645
               IF NOT SETQQ,(A) THEN ERRORTY,(                                   SRC 1646
               ≠LET F OF, X BE Y, F NOT A SET,≠),                                SRC 1647
               DIMF,(A,B),                                                       SRC 1648
               IF C EQ UNDEF, THEN RETURN(A),                                    SRC 1649
               L=GENTUP,(COPY(C)),                                               SRC 1650
               L[3][2]=B;L[3][2],                                                SRC 1651
               L[3][1]=L[3][1]+1,                                                SRC 1652
               L[2]=HASHCOD,(B),                                                 SRC 1653
               AUGMNTS,(A,L)                                                     SRC 1654
            END                                                                  SRC 1655
         END;                                                                    SRC 1656
SETFUNS, = PROC(A,B,C),                                                          SRC 1657
            BEGIN(P,X,L),                                                        SRC 1658
               IF NOT SETQQ,(A) THEN ERRORTY,(                                   SRC 1659
               ≠LET F SOF, X BE Y, F NOT A SET,≠),                               SRC 1660
```

```
            DIMF,(A,B),                                                    SRC 1661
            IF C EQ NULLSET, THEN RETURN(A),                               SRC 1662
            IF NOT SETQQ,(C) THEN ERRORTY,(                                SRC 1663
            ≠LET F SOF, X BE Y, Y NOT A SET,≠),                            SRC 1664
            P=COPY(NILVECT,),X=NEXTELT,(C,P),                              SRC 1665
            WHILE X NE UNDEF, REPEAT DO                                    SRC 1666
               L=GENTUP,(COPY(X)),                                         SRC 1667
               L[3][2]=B;L[3][2],                                          SRC 1668
               L[3][1]=L[3][1]+1,                                          SRC 1669
               L[2]=HASHCOD,(B),                                           SRC 1670
               AUGMNTS,(A,L),                                              SRC 1671
               X=NEXTELT,(C,P)                                             SRC 1672
            END,                                                           SRC 1673
            RETURN(A)                                                      SRC 1674
         END                                                               SRC 1675
      END;                                                                 SRC 1676
                                                                           SRC 1677
SETFUNN, = PROC(A,B,C),                                                    SRC 1678
         BEGIN(K,L,M),                                                     SRC 1679
            IF TL B EQ NIL THEN RETURN(SETSIN,(A,HD B,C));                 SRC 1680
            IF NOT SETQQ,(A) THEN ERRORTY,(                                SRC 1681
            ≠LET F OFN, X BE Y, F NOT A S T,≠),                            SRC 1682
            DIMFN,(A,B),                                                   SRC 1683
            IF C EQ UNDEF, THEN RETURN(A),                                 SRC 1684
            K=B,M=0,                                                       SRC 1685
            WHILE K NE NIL REPEAT DO                                       SRC 1686
               M=M+1, K=TL K                                               SRC 1687
            END,                                                           SRC 1688
            L=GENTUP,(COPY(C)),                                            SRC 1689
            K=VECTOR(TUPL,,HASHCOD,(HD B),                                 SRC 1690
            VECTOR(M,B)),                                                  SRC 1691
            K=TPLUS,(K,L),                                                 SRC 1692
            AUGMNTS,(A,K),                                                 SRC 1693
            RETURN(A)                                                      SRC 1694
         END                                                               SRC 1695
      END;                                                                 SRC 1696
                                                                           SRC 1697
STFUNNS,=PROC(A,B,C),                                                      SRC 1698
         BEGIN(P,X,K,L,M),                                                 SRC 1699
            IF TL B EQ NIL THEN RETURN(SETFUNS,(A,HD B,C)),                SRC 1700
            IF NOT SETQQ,(A) THEN ERRORTY,(                                SRC 1701
            ≠LET F SOFN, X BE Y, F NOT A SET,≠),                           SRC 1702
            DIMFN,(A,B),                                                   SRC 1703
            IF NOT SETQQ,(C) THEN ERRORTY,(                                SRC 1704
            ≠LET F SOFN, X BE Y, Y NOT A SET,≠),                           SRC 1705
            IF C EQ NULLSET, THEN RETURN(A),                               SRC 1706
            P=COPY(NILVECT,), X=NEXTELT,(C,P),                             SRC 1707
            K=B,M=0,                                                       SRC 1708
            WHILE K NE NIL REPEAT DO                                       SRC 1709
               M=M+1, K=TL K                                               SRC 1710
            END,                                                           SRC 1711
            WHILE X NE UNDEF, REPEAT DO                                    SRC 1712
               L=GENTUP,(COPY(X)),                                         SRC 1713
               K=VECTOR(TUPL,,HASHCOD,(HD B),                              SRC 1714
               VECTOR(M,B)),                                               SRC 1715
               K=TPLUS,(K,L),                                              SRC 1716
               AUGMNTS,(A,K),                                              SRC 1717
               X=NEXTELT,(C,P)                                             SRC 1718
            END,                                                           SRC 1719
            RETURN(A)                                                      SRC 1720
```

```
            END                                                                 SRC 1721
         END;                                                                   SRC 1722
                                                                                SRC 1723
                                                                                SRC 1724
COMMENT ≠ SINISTER CALLS FOR HEAD. AND TAIL.≠                                   SRC 1725
HEADSIN. = PROC(T,X),                                                           SRC 1726
            BEGIN(TT),                                                          SRC 1727
               IF NOT TUPQ.(T) THEN ERRORTY.(                                   SRC 1728
               ≠LET HEAD. T BE Q. T NOT A TUPLE.≠),                             SRC 1729
               IF T EQ NULT. THEN DO                                            SRC 1730
                  TT=GENTUP.(COPY(X)),                                          SRC 1731
                  T[2]=TT[2],                                                   SRC 1732
                  T[3]=TT[3],                                                   SRC 1733
                  RETURN(T)                                                     SRC 1734
               END,                                                             SRC 1735
               T[2]=HASHCOD.(X),                                                SRC 1736
               HD (T[3][2]) = COPY(X),                                          SRC 1737
               RETURN(T)                                                        SRC 1738
            END                                                                 SRC 1739
         END;                                                                   SRC 1740
TAILSIN.=  PROC(T,X),                                                           SRC 1741
            BEGIN(),                                                            SRC 1742
               IF NOT TUPQ.(T) THEN ERRORTY.(                                   SRC 1743
               ≠LET TAIL. T BE Q. T NOT A TUPLE.≠),                             SRC 1744
               IF T EQ NULT. THEN ERRORVA.(                                     SRC 1745
               ≠LET TAIL. T BE Q. T IS NULL TUPLE.≠),                           SRC 1746
               TL(T[3][2])=IF NOT TUPQ.(X) THEN LIST(COPY(X))                   SRC 1747
               ELSE COPY(TUPLE.(X)),                                            SRC 1748
               RETURN(T)                                                        SRC 1749
            END                                                                 SRC 1750
         END;                                                                   SRC 1751
                                                                                SRC 1752
FROMSET.=  PROC(S),                                                             SRC 1753
            BEGIN(X),                                                           SRC 1754
            IF NOT SETQQ.(S) THEN ERRORTY.(                                     SRC 1755
            ≠A FROM S. S NOT A SET.≠),                                          SRC 1756
            IF S EQ NULLSET. THEN ERRORVA.(                                     SRC 1757
            ≠A FROM S. S IS NULL SET.≠),                                        SRC 1758
            X=ARB.(S),                                                          SRC 1759
            DIMINIS.(S.X),                                                      SRC 1760
            RETURN(X)                                                           SRC 1761
            END                                                                 SRC 1762
         END;                                                                   SRC 1763
                                                                                SRC 1764
COMMENT ≠ STRINGF. CONVERTS A SETL-OBJECT INTO A CHARACTER-STRING ≠             SRC 1765
STRINGF.= PROC(X),                                                              SRC 1766
            IF INTQ(X) THEN DEC.(X)                                             SRC 1767
            ELSEIF STRQ(X) THEN X                                               SRC 1768
            ELSEIF X EQ TRUE THEN ≠TRUE.≠                                       SRC 1769
            ELSEIF X EQ FALSE THEN ≠FALSE.≠                                     SRC 1770
            ELSEIF BLANKQ.(X) THEN CONCAT(CONCAT(≠;BLANK≠.DEC.(X[3])            SRC 1771
                ).≠;≠)                                                          SRC 1772
            ELSEIF LBLQ(X) THEN CONCAT(CONCAT(≠;LABEL≠.                         SRC 1773
                SFROMI.(IFROMID(X).10)).≠;≠)                                    SRC 1774
            ELSEIF CODEQ(X) THEN CONCAT(CONCAT(≠/SUBR ≠.                        SRC 1775
                SFROMID(IDFROMC(X))).≠/≠)                                       SRC 1776
            ELSEIF X EQ UNDEF. THEN ≠**OM.**≠                                   SRC 1777
            ELSEIF SETQQ.(X) THEN SFROMST.(X)                                   SRC 1778
            ELSEIF TUPQ.(X) THEN SFROMTP.(X)                                    SRC 1779
            ELSE ≠*******≠                                                      SRC 1780
```

```
          END;                                                                  SRC 1784
                                                                                SRC 1785
 SFROMST,= PROC(X),                                                             SRC 1786
             BEGIN(C,Z,P),                                                      SRC 1787
                IF X EQ NL, THEN RETURN(≠NL,≠),                                 SRC 1788
                P=COPY(NILVECT,), Z=NEXTELT,(X,P),                              SRC 1789
                C=CONCAT(≠≤:≠,STRINGF,(Z)),                                     SRC 1790
                Z=NEXTELT,(X,P),                                                SRC 1791
                WHILE Z NE UNDEF, REPEAT DO                                     SRC 1792
                   C=CONCAT(CONCAT(C,≠,≠),STRINGF,(Z)),                         SRC 1793
                   Z=NEXTELT,(X,P)                                              SRC 1794
                END,                                                            SRC 1795
                C=CONCAT(C,≠≥≠),                                                SRC 1796
                RETURN(C)                                                       SRC 1797
             END                                                                SRC 1798
          END;                                                                  SRC 1799
                                                                                SRC 1800
 SFROMTP,= PROC(X),                                                             SRC 1801
             BEGIN(C,Z,P),                                                      SRC 1802
                IF X EQ NULT, THEN RETURN(≠NULT,≠),                             SRC 1803
                P=COPY(NILVECT,), Z=NXTLTT2,(X,P),                              SRC 1804
                C=CONCAT(≠<≠,STRINGF,(Z)),                                      SRC 1805
                Z=NXTLTT2,(X,P),                                                SRC 1806
                WHILE P[1] NE UNDEF, REPEAT DO                                  SRC 1807
                   C=CONCAT(CONCAT(C,≠,≠),STRINGF,(Z)),                         SRC 1808
                   Z=NXTLTT2,(X,P)                                              SRC 1809
                END,                                                            SRC 1810
                C=CONCAT(C,≠>≠),                                                SRC 1811
                RETURN(C)                                                       SRC 1812
             END                                                                SRC 1813
          END;                                                                  SRC 1814
*****    THIS IS A BALM FIX                                                     SEPT7163
 MSYMB,=VECTOR(SLASHX,,DNUM,[1],DNUM,[1],DNUM,[1],DNUM,[1]);                    SEPT7164
 GENNAM,= PROC(),                                                               SEPT7165
        BEGIN(I,J),                                                             SEPT7166
        FOR I=(5,2,-1) REPEAT                                                   SEPT7167
          IF (J=CHAR,[MSYMB,[I]]) LT 9 THEN DO                                  SEPT7168
             MSYMB,[I]=DNUM,[J+2], RETURN IDFROMS(SFROMV(MSYMB,))               SEPT7169
          END                                                                   SEPT7170
        ELSE MSYMB,[I]=DNUM,[1]                                                 SEPT7171
        END                                                                     SEPT7172
 END;                                                                           SEPT7173
***** END OF THE BALM FIX                                                       SEPT7174
*EXTERNAL NELT DIFFERENT FROM NELT, IN THAT IT IS AN ERROR WHEN X IS             SEPT7175
*         AN ATOM                                                               SEPT7176
 NELT=    PROC(X),                                                              SEPT7177
             IF STRQ(X) THEN SIZE(X)                                            SEPT7178
             ELSEIF SETQQ,(X) OR TUPQ,(X) THEN X[3][1]                          SEPT7179
             ELSEIF BSTRQ,(X) THEN BLEN(X)                                      SEPT7180
             ELSEIF X EQ UNDEF, THEN ERRORVA,(≠NELT X, X IS UNDEFINED≠)         SEPT7181
             ELSE ERRORTY,(≠NELT X, X NOT A LEGAL TYPE≠)                        SEPT7182
       END;                                                                     SEPT7183
*      DEBUGGING PROCEDURES                                                     SEPT7203
                                                                                SEPT7204
*    THESE INCLUDE ERROR, SAVESETL AND CRASH                                    SEPT7205
*   ALL OF THERE MUST BE IN BALMSETL EVEN IF THE DEBUGGING PROCEDURES ARE       SEPT7206
*   REMOVED                                                                     SEPT7207
 ATEXIT=PROC(LINE,P),                                                           SEPT7208
       BEGIN($PUTITEM,),                                                        DEC72  5
          IF ¬ATEXTRC THEN RETURN NIL,                                          SEPT7210
```

```
         PUTITEM,=PUT,,                                                    SEPT7211
         PRINT(MINUS,,MINUS,,MINUS,,=RETURN,=FROM,P,=AT,LINE)              SEPT7212
      END                                                                  SEPT7213
   END;                                                                    SEPT7214
ATEXITV=PROC(LINE,P),                                                      SEPT7215
      BEGIN(),                                                             SEPT7216
         IF ¬ATEXTRC THEN RETURN NIL,                                      SEPT7217
         PRINT(MINUS,,MINUS,,MINUS,,=RETURN,                               SEPT7218
                  =FROM,P,=AT,LINE,=WITH, =VALUE, ATEXVAL)                 SEPT7219
      END                                                                  SEPT7220
   END;                                                                    SEPT7221
ATSETVSN= PROC(LINE,P,NAM,VAL),                                            SEPT7222
      IF ATEQTRC THEN                                                      SEPT7223
     PRINT(MINUS,,MINUS,,MINUS,,=AT,LINE,=IN,P,NAM,=IS,VAL)                SEPT7224
   END;                                                                    SEPT7225
ATASSERT= PROC(A,B,C,D),                                                   SEPT7226
      DO PRINT(≠ASSERTION FAILED≠), ATSETLSN(A,B,C,D) END                  SEPT7227
   END;                                                                    SEPT7228
ATSN=PROC(N,P),                                                            SEPT7229
         BEGIN($PUTITEM,,L),                                               DEC72  6
         PUTITEM,=PUT,,                                                    SEPT7231
         L=SIZE ATSNS,                                                     SEPT7232
         IF ATSNTRC THEN                                                   SEPT7233
       PRINT(MINUS,,MINUS,,MINUS,,=LINE,N,MINUS,,=IN,P);                   SEPT7234
         ATSNP=ATSNP+1,                                                    SEPT7235
         ATSNS[ATSNP=((ATSNP-1)/ L)* L]=N;P                                SEPT7236
   END END;                                                                SEPT7237
SAVESETL=PROC(),                                                           SEPT7238
      BEGIN(FN,SAV),                                                       NOV72  2
         SAV=TRACE,                                                        NOV72  3
         FN=OPEN(≠TAPE9≠,80),  REWIND(FN),                                 SEPT7240
         TRACE=1, GARBCOLL(), SAVEALL(FN),                                 SEPT7241
         TRACE=SAV, CLOSE(≠BLM4SVD≠),                                      NOV72  4
         PRINT(≠BALMSETL,CY=67 SAVED ON TAPE9≠),                           JAN74  4
         ATSNS=MAKVECTO(10), ATSNP=0                                       SEPT7245
      END                                                                  SEPT7246
   END;                                                                    SEPT7247
ATSNS=MAKVECTO(10);                                                        SEPT7248
ATSNP=0;                                                                   SEPT7249
ATSNTRC=FALSE;                                                             SEPT7250
ATEQTRC=TRUE;                                                              SEPT7251
ATEXTRC=TRUE;                                                              SEPT7252
ATSETV=PROC(NAM,VAL), NAM END;                                             SEPT7253
ATSETL=PROC(NAMLIST, VALLIST), NAMLIST END;                                SEPT7254
ATENTRY=PROC(P),                                                           SEPT7255
         BEGIN($PUTITEM,),                                                 DEC72  8
         IF ¬ATEXTRC THEN RETURN NIL,                                      SEPT7257
         PUTITEM,=PUT,, PRINT(MINUS,,MINUS,,MINUS,,=ENTERING,P)            SEPT7258
      END                                                                  SEPT7259
   END;                                                                    SEPT7260
ATSETLSN= PROC(LINE,P,NAMLIST,VALLIST),                                    SEPT7261
      IF ATEQTRC THEN DO                                                   SEPT7262
         PRINT(MINUS,,MINUS,,MINUS,,=AT,LINE,=IN,P),                       SEPT7263
         WHILE PAIRQ(NAMLIST) REPEAT DO                                    SEPT7264
         PRINT(SPCR,,HD NAMLIST,=IS,HD VALLIST),                           NOV72  5
         NAMLIST=TL NAMLIST, VALLIST=TL VALLIST                            SEPT7266
      END END                                                              SEPT7267
   END;                                                                    SEPT7268
ERROR =  PROC(TY),                                                         SEPT7269
      DO ATSNLIST(),                                                       SEPT7270
```

```
          IF TY EQ 3 THEN DO                                                    SEPT7271
          PROCTRAC(TL STKTRACE(LEVEL,)), EXECUTE(INPUT,OUTPUT);                 DEC72  9
          STOP() END                                                            SEPT7273
          ELSE STOP()                                                           SEPT7274
       END                                                                      SEPT7275
    END;                                                                        SEPT7276
LEVEL,=10; COMMENT ≠LEVEL OF BACKTRACE IN CASE OF CRASH≠                        DEC72 10
CRASH, =  PROC(),                                                               SEPT7278
       BEGIN(L),                                                                SEPT7279
          PRINT(≠*** CRASH ***≠),                                               SEPT7280
          ATSNLIST(),                                                           SEPT7281
          L=STKTRACE(LEVEL,+2), PROCTRAC(TL TL L),                              DEC72 11
          CRSHCNT,=CRSHCNT,+1,                                                  SEPT7283
          IF CRSHCNT, LT CRASHMAX THEN                                          SEPT7284
          EXECUTE(INPUT,OUTPUT),                                                SEPT7285
          STOP()                                                                SEPT7286
       END                                                                      SEPT7287
    END;                                                                        SEPT7288
ECHO=TRUE;                                                                      NOV72  6
READIN,   = PROC(),                                                             NOV72  7
       BEGIN(),                                                                 NOV72  8
START,    LIN=RDLINE(FN),                                                       NOV72  9
          IF ECHO AND FN EQ 1 THEN                                              NOV72 10
             WRLINE(CONCAT(STRING(BLANK,),LIN),OUTPUT[1]),                      NOV72 11
          LIN=VFROMS(LIN),                                                      NOV72 12
          IF (LLEN=SIZE LIN) GT 0 AND LIN[1] EQ STAR, THEN GOTO START,          NOV72 13
          NEXT=1                                                                NOV72 14
       END                                                                      NOV72 15
    END;                                                                        NOV72 16
SPCR,=IDFROMS(≠     ≠);                                                         NOV72 17
PROCTRAC= PROC(L),                                                              NOV72 18
       BEGIN(V,N,I,NAM),                                                        NOV72 19
          WRLINE(≠ PROCEDURE TRACE - IN REVERSE CALLING SEQUENCE≠,              NOV72 20
             OUTPUT[1]),                                                        NOV72 21
          WHILE L AND(NAM=FINDNAM,(IDFROMC(HD HD L)))REPEAT                     NOV72 22
          DO                                                                    NOV72 23
             PRINT,(NAM), V=TL HD L, N=SIZE V,                                  NOV72 24
             FOR I=(1,N) REPEAT                                                 NOV72 25
             PRINT,(SPCR,,=ARG,I,==,V[I]),                                      NOV72 26
             L= TL L, IF NAM EQ =MAIN THEN L=NIL                                NOV72 27
          END                                                                   NOV72 28
       END                                                                      NOV72 29
    END;                                                                        NOV72 30
OPLIST,=LIST(=IDFROMI,41B);OPLIST,;                                             NOV72 31
FINDNAM,=PROC(NAM),                                                             NOV72 32
       BEGIN(I,N),                                                              NOV72 33
          IF NAM EQ =CURCOM, THEN RETURN =MAIN                                  NOV72 34
          ELSEIF EQSTR(SUB(SFROMID(NAM),1,1),≠*≠) THEN                          NOV72 35
             DO N=IFROMID(NAM)-1,                                               NOV72 36
             FOR I=(N,1,-1) REPEAT                                              NOV72 37
             IF VALUE(IDFROMI(I)) EQ VALUE NAM THEN RETURN IDFROMI(I),          NOV72 38
             RETURN NIL                                                         NOV72 39
             END                                                                NOV72 40
          ELSE RETURN NAM                                                       NOV72 41
       END                                                                      NOV72 42
    END;                                                                        NOV72 43
OPLIST,=TL OPLIST,;                                                             NOV72 44
ATSNLIST=PROC(),                                                                SEPT7289
          BEGIN(X,I,L,$PUTITEM,),                                               DEC72 12
          PUTITEM,=PUT,,                                                        SEPT7291
```

```
        FOR I≡(1,L=SIZE ATSNS) REPEAT                                   SEPT7292
          IF PAIRQ(ATSNS[I]) THEN GOTO LBL.                             SEPT7293
        RETURN NIL.                                                     SEPT7294
LBL.  PRINT(≠LAST STATEMENTS EXECUTED≠).                                SEPT7295
      PRINT(=LINE,=PROC).                                               SEPT7296
        FOR I≡(ATSNP+1,ATSNP+L) REPEAT                                  SEPT7297
        IF PAIRQ(X=ATSNS[I-((I-1)/ L)* L])THEN PRINT(HD X,TL X)         SEPT7298
  END END;                                                              SEPT7299

CHGCHAR(≡/,13);  COMMENT≠ ALLOW // AS AN OPERATOR≠                      DEC72 13
COMMENT ≠ OPERATOR DECLARATIONS ≠                                       DEC72 14
INFIX(=EL,710,710,=ELTF,);                                              DEC72 15
INFIX(=→,710,710,=ELTF,);                                               DEC72 16
UNARY(=PAIRTUP,2000,PAIRTUP,);                                          DEC72 17
UNARY(=TYPE,2000,=TYPE,);                                               DEC72 18
INFIX(=+,1501,1500,=PLS,);                                              DEC72 19
INFIX(=-,1501,1500,=MNS,);                                              DEC72 20
INFIX(=*,1601,1600,=TMS,);                                              DEC72 21
INFIX(=DSLSH,1601,1600,=DSLSH,);                                        DEC72 22
INFIX(=/,1601,1600,=SLSH,);                                             DEC72 23
INFIX(=//,1601,1600,=DSLSH,);                                           DEC72 24
INFIX(=WITH,751,750,=WITH,);INFIX(=LESS,751,750,=LESS,);                DEC72 25
UNARY(=HEAD,2000,=HEAD,);                                               DEC72 26
UNARY(=TAIL,2000,=TAIL,);                                               DEC72 27
INFIX(=TAILN,750,750,=TAILN,);                                          DEC72 28
UNARY(=ARB,2000,=ARB,);                                                 DEC72 29
INFIX(=INCS,751,750,=INCS,);                                            DEC72 30
INFIX(=EQUAL,1400,1400,=EQUALT,);                                       DEC72 31
INFIX(=NEQUAL,1400,1400,=NEQUALT,);                                     DEC72 32
UNARY(=DFD,2000,=NOMEGAP,);                                             DEC72 33
UNARY(=UNDFD,2000,=OMEGAP,);                                            DEC72 34
UNARY(=ATOM,2000,=ATOM,);                                               DEC72 35
UNARY(=ABS,2000,=ABS,);                                                 DEC72 36
UNARY(=NELT,2000,=NELT);                                                DEC72 37
UNARY(=↓,2000,=NELT);                                                   DEC72 38
UNARY(=DEC,2000,=DEC,);                                                 DEC72 39
UNARY(=OCT,2000,=OCT,);                                                 DEC72 40
INFIX(=MAX,1851,1850,=MAX,);                                            DEC72 41
INFIX(=MIN,1851,1850,=MIN,);                                            DEC72 42
INFIX(=LESF,750,750,=LESF,);                                            DEC72 43
INFIX(=COMP,750,750,=SELTUPL,);                                         DEC72 44
INFIX(=LESFN,750,750,=LESFN,);                                          DEC72 45
                                                                        DEC72 46
COMMENT ≠ ,,, OPERATOR DECLARATIONS FOR FUNCTIONAL APPLICATION ,,, ≠     DEC72 47
                                                                        DEC72 48
INFIX(=OF,750,751,=OF,);                                                DEC72 49
INFIX(=SOF,750,751,=SOF,);                                              DEC72 50
INFIX(=BOF,750,751,=BOF,);                                              DEC72 51
INFIX(=OFN,750,751,=OFN,);                                              DEC72 52
INFIX(=SOFN,750,751,=SOFN,);                                            DEC72 53
INFIX(=BOFN,750,751,=BOFN,);                                            DEC72 54
                                                                        DEC72 55
COMMENT ≠ ,,, MACRO DEFINITIONS ,,, ≠                                   DEC72 56
                                                                        DEC72 57
COMMENT ≠ DEFINE OPERATORS USED IN MACROS ≠                             DEC72 58
                                                                        DEC72 59
UNARY(=LET,740,=LET);                                                   DEC72 60
INFIX(=BE,690,690,=BE);                                                 DEC72 61
UNARY(=FORALL,590,=FORALL);                                             DEC72 62
UNARY(=EXISTS,590,=EXISTS);                                             DEC72 63
                                                                        DEC72 64
```

```
INFIX(=SUCH,605,605,=SUCH);                                          DEC72 65
INFIX(=IN,600,600,=IN);                                              DEC72 66
INFIX(=IS,700,701,=IS);                                              DEC72 67
INFIX(=QUANT,605,605,=QUANT);                                        DEC72 68
INFIX(=INR,600,600,=INR);                                            DEC72 69
INFIX(=FR,500,500,=FROM);                                            DEC72 70
                                                                     DEC72 71
COMMENT ≠ MAKVLOCAL IS USED TO ELIMINATE VERY LOCAL VARIABLES FROM   DEC72 72
          THE SYSTEM AT THE USER LEVEL.....≠                         DEC72 73
MAKVLOCA(NIL);                                                       DEC72 74
                                                                     DEC72 75
COMMENT ≠ USE THE ,,MEANS,, STATEMENT TO DEFINE THE MACROS,          DEC72 76
          THE VARIABLES MUST BE X1,X2,X3, ETC.,, ≠                   DEC72 77
                                                                     DEC72 78
FORALL X1 EL X2 SUCH X3 REPEAT X4                                    DEC72 79
MEANS                                                                DEC72 80
DO                                                                   DEC72 81
       G1=COPY(NILVECT,),                                            DEC72 82
       X1=NEXTELT,(X2,G1),                                           DEC72 83
       WHILE X1 NE UNDEF, REPEAT DO                                  DEC72 84
            IF X3 THEN X4,                                           DEC72 85
            X1=NEXTELT,(X2,G1)                                       DEC72 86
       END                                                           DEC72 87
END;                                                                 DEC72 88
FORALL X1 EL X2 REPEAT X4                                            DEC72 89
MEANS                                                                DEC72 90
DO                                                                   DEC72 91
       G1=COPY(NILVECT,),                                            DEC72 92
       X1=NEXTELT,(X2,G1),                                           DEC72 93
       WHILE X1 NE UNDEF, REPEAT DO                                  DEC72 94
            X4, X1=NEXTELT,(X2,G1)                                   DEC72 95
       END                                                           DEC72 96
END;                                                                 DEC72 97
                                                                     DEC72 98
FORALL X1 IN(X2,X3) SUCH X4 REPEAT X5                                DEC72 99
MEANS                                                                DEC72100
   FOR X1=(X2,X3,1) REPEAT IF X4 THEN X5                             DEC72101
;                                                                    DEC72102
FORALL X1 IN(X2,X3) REPEAT X4                                        DEC72103
MEANS                                                                DEC72104
   FOR X1=(X2,X3,1) REPEAT X4                                        DEC72105
;                                                                    DEC72106
FORALL X1 INR(X2,X3) SUCH X4 REPEAT X5                               DEC72107
MEANS                                                                DEC72108
   FOR X1=(X2,X3,-1) REPEAT IF X4 THEN X5                            DEC72111
;                                                                    NOV73 12
FORALL X1 INR(X2,X3) REPEAT X4                                       DEC72113
MEANS                                                                DEC72114
   FOR X1=(X2,X3,-1) REPEAT X4                                       DEC72117
;                                                                    NOV73 13
COMMENT ≠ ,,, PREDICATES ,,, ≠                                       DEC72119
EXISTS X1 EL X2 SUCH X3                                              DEC72120
MEANS                                                                DEC72121
BEGIN(MACVAR,),                                                      DEC72122
   IF NELT,(X2) EQ 0 THEN RETURN(FALSE),                             DEC72123
   MACVAR,=COPY(NILVECT,), X1=NEXTELT,(X2,MACVAR,),                  DEC72124
   WHILE X1 NE UNDEF, REPEAT DO                                      DEC72125
      IF X3 THEN RETURN(TRUE), X1=NEXTELT,(X2,MACVAR,)               DEC72126
   END,                                                              DEC72127
   RETURN(FALSE)                                                     DEC72128
```

```
END;                                                                    DEC72129
EXISTS X1 IN(X2,X3) SUCH X4                                             DEC72130
MEANS                                                                   DEC72131
BEGIN(MACVAR,),                                                         DEC72132
   MACVAR,=IF X3 GE X2 THEN 1 ELSE -1,                                  DEC72133
   FOR X1=(X2,X3,MACVAR,) REPEAT IF X4 THEN RETURN(TRUE),               DEC72134
   RETURN(FALSE)                                                        DEC72135
END;                                                                    DEC72136
FORALL X1 EL X2 QUANT X3                                                DEC72137
MEANS                                                                   DEC72138
BEGIN(MACVAR,),                                                         DEC72139
   IF NELT,(X2) EQ 0 THEN RETURN(TRUE),                                 DEC72140
   MACVAR,=COPY(NILVECT,), X1=NEXTELT,(X2,MACVAR,),                     DEC72141
   WHILE X1 NE UNDEF, REPEAT DO                                         DEC72142
      IF NOT(X3) THEN RETURN(FALSE), X1=NEXTELT,(X2,MACVAR,)            DEC72143
   END,                                                                 DEC72144
   RETURN(TRUE)                                                         DEC72145
END;                                                                    DEC72146
FORALL X1 IN (X2,X3) QUANT X4                                           DEC72147
MEANS                                                                   DEC72148
BEGIN(MACVAR,),                                                         DEC72149
   MACVAR,=IF X3 GE X2 THEN 1 ELSE -1,                                  DEC72150
   FOR X1=(X2,X3,MACVAR,) REPEAT IF NOT(X4) THEN RETURN(FALSE),         DEC72151
   RETURN(TRUE)                                                         DEC72152
END;                                                                    DEC72153
                                                                        DEC72154
COMMENT ≠ ,,, OTHER MACROS ,,, ≠                                        DEC72155
X1 IS X2   MEANS   X2 = X1  ;                                           DEC72156
X1 FR X2 MEANS   X1=FROMSET,(X2)  ;                                     DEC72157
COMMENT ≠ MACROS FOR SINISTER CALLS ≠                                   DEC72158
LET X1 COMP X2 BE X3 MEANS SETSIN,(X1,X2,X3);                           DEC72159
LET X1 OF X2 BE X3 MEANS SETSIN,(X1,X2,X3);                             DEC72160
LET X1 OFN X2 BE X3   MEANS   SETFUNN,(X1,X2,X3);                       DEC72161
LET X1 SOFN X2 BE X3   MEANS   STFUNNS,(X1,X2,X3);                      DEC72162
LET X1 SOF X2 BE X3   MEANS   SETFUNS,(X1,X2,X3);                       DEC72163
LET HEAD X1 BE X2   MEANS   HEADSIN,(X1,X2);                            DEC72164
LET TAIL X1 BE X2   MEANS   TAILSIN,(X1,X2);                            DEC72165
COMMENT ≠ GENERAL SET,=FORMER ≠                                         DEC72166
UNARY(=SETOF,590,=SETOF);                                               DEC72167
INFIX(=WHERE,610,610,=WHERE);                                           DEC72168
SETOF X1 WHERE X2 EL X3                                                 DEC72169
MEANS                                                                   DEC72170
BEGIN(MACVAR,,MACVAR1),                                                 DEC72171
   MACVAR,=COPY(NILVECT,), MACVAR1=COPY(NL,),                           DEC72172
   X2=NEXTELT,(X3,MACVAR,),                                             DEC72173
   WHILE X2 NE UNDEF, REPEAT DO                                         DEC72174
      AUGMNTS,(MACVAR1,X1),                                             DEC72175
      X2=NEXTELT,(X3,MACVAR,)                                           DEC72176
   END,                                                                 DEC72177
   RETURN(MACVAR1)                                                      DEC72178
END;                                                                    DEC72179
SETOF X1 WHERE X2 EL X3 SUCH X4                                         DEC72180
MEANS                                                                   DEC72181
BEGIN(MACVAR,,MACVAR1),                                                 DEC72182
   MACVAR,=COPY(NILVECT,), MACVAR1=COPY(NL,),                           DEC72183
   X2=NEXTELT,(X3,MACVAR,),                                             DEC72184
   WHILE X2 NE UNDEF, REPEAT DO                                         DEC72186
      IF X4 THEN AUGMNTS,(MACVAR1,X1),                                  DEC72187
      X2=NEXTELT,(X3,MACVAR,)                                           DEC72188
   END,                                                                 DEC72189
```

```
   RETURN(MACVAR1)                                                          DEC72190
END;                                                                        DEC72191
SETOF X1 WHERE X2 IN(X3,X4)                                                 DEC72192
MEANS                                                                       DEC72193
BEGIN(MACVAR,,MACVAR1),                                                     DEC72194
   MACVAR,=IF X4 GE X3 THEN 1 ELSE -1, MACVAR1=COPY(NL,),                   DEC72195
   FOR X2=(X3,X4,MACVAR,) REPEAT AUGMNTS.(MAC AR1,X1),                      DEC72196
   RETURN(MACVAR1)                                                          DEC72197
END;                                                                        DEC72198
SETOF X1 WHERE X2 IN(X3,X4) SUCH X5                                         DEC72199
MEANS                                                                       DEC72200
BEGIN(MACVAR,,MACVAR1),                                                     DEC72201
   MACVAR,=IF X4 GE X3 THEN 1 ELSE -1, MACVAR1=COPY(NL,),                   DEC72202
   FOR X2=(X3,X4,MACVAR,) REPEAT IF X5 THEN AUGMNTS.(MACVAR1,X1),           DEC72203
   RETURN(MACVAR1)                                                          DEC72204
END;                                                                        DEC72205
                                                                            DEC72206
                                                                            DEC72207
COMMENT ≠MAKE PROPERTY-LISTS OUT OF. OPERATOR LISTS FOR EFFICIENCY ≠        DEC72208
MAKPROPS(=INFIX,=INFIXLI,);                                                 DEC72209
MAKPROPS(=MACRO,=MACROLI,);                                                 DEC72210
MAKPROPS(=UNARY,=UNARYLI,);                                                 DEC72211
DUP,(=BLANKZZ,,=SETZZ,,=TUPLZZ,,=INTZZ,,=SUBRZZ,,=LABZZ,,=STRZZ,,           SRC 2039
    =BITSZZ,);                                                              SRC 2040
DUP,(=IDENTQ,,=APPEND,,=PRT,,=UNDEF,,=NULLSET,, =NULC,                      SRC 2041
=NULB,,=NILVECT,,=TUPQ,,=SETQQ,,=BLANKQ,,=BSTRQ,,=OMEGAP,,=NOMEGAP,,       SRC 2042
=SIZ,,=AUGMENT,,=DIMF,,=UNION,,=SYMDIF,,=NEWAT,,                           SRC 2043
=GENSET,,=GENTUP,,=POW,,=NPOW,,=GETWD,,=SKIPWD,,=FIRSTWD,,=NEXTELT,,        SRC 2044
=DUP,,=DUPL,,=FROMSET,,=MACVAR,);                                           SRC 2045
                                                                            SRC 2046
DUP,(=DIMFN,,=RANDOM,);                                                     SRC 2047
DIMINISH = DIMINIS,; INTERSCT = INTRSCT,;                                   SEPT7300
TAILSPEC=TAILSRC,;   STRINGOF= STRINGF,;                                    SRC 2049
NULLTUPL = NULLTUP,;                                                        SRC 2050
CHGCHAR(=,,11);                                                             SRC 2051
DO SAVESETL(),                                                              NOV72 45
ECHO=NIL,                                                                   NOV72 46
PRINT(≠DATE SAVED JANUARY 1974≠)                                            JAN74  5
END;                                                                        NOV72 48
```