

H A S P J O B L O G

```
$14.37.41 JOB 109 -- RICK -- BEGINNING EXEC - INIT 1 - CLASS B
14.37.44 JOB 109 IEF244I RICK . .TAPE UNABLE TO ALLOCATE
14.37.44 JOB 109 IEF388I RICK . .TAPE WAITING FOR DEVICES
14.37.52 JOB 109 IEF233A M OC1.TAPE .,RICK
14.38.46 JOB 109 IEF280E K OC1.TAPE ,RICK
$14.38.48 JOB 109 END EXECUTION.
```

---- HASP-II JOB STATISTICS ----

6 CARDS READ

2,723 SYSOUT PRINT RECORDS

0 SYSOUT PUNCH RECORDS

6.50 SECONDS EXECUTION TIME

//RICK JOB COMP.VALENTE1641,MSGLEVEL=(1,1),CLASS#B

JOB 109

***JOBPARM L=10

// EXEC PGM=TAPEDUMP

//SYSPRINT DD SYSOUT=A

//TAPE DD DSN=TAPE,DISF=OLD,VOL=SER=TAPE,LABEL=(,BLP),UNIT=TAPE

//

IEF244I RICK . .TAPE UNABLE TO ALLOCATE

IEF388I RICK . .TAPE WAITING FOR DEVICES

IEF236I ALLOC. FOR RICK

IEF237I 623 ALLOCATED TO SYSPRINT

IEF237I 0C1 ALLOCATED TO TAPE

IEF142I - STEP WAS EXECUTED - COND CODE 0000

IEF285I TAPE KEPT

IEF285I VOL SER NOS= TAPE .

IEF280E K 0C1,TAPE ,RICK

IEF373I STEP / / START 77021.1437

IEF374I STEP / / STOP 77021.1438 CPU 0MIN 06.50SEC MAIN 66K LCS 0K

IEF375I JOB /RICK / START 77021.1437

IEF376I JOB /RICK / STOP 77021.1438 CPU 0MIN 06.50SEC

UNLABELED TAPE

1 1
1 2
1 3
1 4
1 5
1 6
1 7
1 8
1 9
1 10
1 11
1 12
1 13
1 14
1 15
1 16
1 17
1 18
1 19
1 20
1 21
1 22
1 23
1 24
1 25
1 26
1 27
1 28
1 29
1 30
1 31
1 32
1 33
1 34
1 35
1 36
1 37
1 38
1 39
1 40
1 41
1 42
1 43
1 44
1 45
1 46
1 47
1 48
1 49
1 50
2 1
2 2
2 3
2 4
2 5
2 6
2 7
2 8

```

C*****LIS00010
C*                                     *LIS00020
C*                                     *LIS00030
C*           LISP F2                   *LIS00040
C*           INTERPRETER               *LIS00050
C*                                     *LIS00060
C*           WRITTEN BY  DAVID W. BRAY AND *LIS00070
C*           RICHARD D. VALENTE        *LIS00080
C*           CLARKSON COLLEGE OF TECHNOLOGY *LIS00090
C*                                     *LIS00100
C*                                     LIS00110
C*           THIS IS A MODIFIED VERSION OF LISP F1  ORIGINALLY WRITTEN BY
C*           MATS NORDSTROM AND        *LIS00120
C*           ERIK SANDEWALL           *LIS00130
C*           DEPARTMENT OF COMPUTER SCIENCE *LIS00140
C*           STUREGATAN 4 B          LIS00150
C*           752 23 UPPSALA  SWEDEN    *LIS00160
C*                                     *LIS00170
C*           VERSION DATE  -  FEB. 1, 1974  LIS00180
C*                                     *LIS00190
C*****LIS00200
C*           DIMENSION LERR(3)         LIS00210
C*           COMMON AND INTEGER DECLARATIONS LIS00220
C-----LIS00230
C           COMMON  IP,      JP,      IBR,      IPP,      JPP,      ATENDA,
C           NFREEP,  NFREEB,  LASTR,   ARG,      ARG2,   ARG3,
C           ARG4,   MODE,    ALIST,   SUBR0,   SUBR1,  SUBR2,
C           SUBR3,  A2,      A8,     A9,     LAMBDA, LABEL,
C           FUNARG, NIL,     EXPR,   FEXPR,  SUBR,   FSUBR,
C           T,     SYS,     RDPNT,  PRTPNT, BSTACK(5),
C           DREG(24), APVAL
C           COMMON /CONST/ NATOM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,
C           LUNUT,IBIG,IRANGE
C           COMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC,
C           IPLUS,IMINUS,IFIG(10),ATEND,ALLSP,CC
C           COMMON/CARSTR/NILCAR(2),CAR(19000)
C           COMMON/CDRSTR/NILCDR(2),CDR(19000)
C           COMMON/ATOM1/ PNAME(8192)
C           COMMON/ATOM2/ PNAME1,PNAME2
C           LOGICAL*1 PNAME
C           COMMON /STKSTR/ STACK(950)
C           COMMON /INITCM/ ITYPAT(8)
C           INTEGER PNAME1,  PNAME2,  SPACE,  RPAR,  COMMA,  DOT,
C           ALLSP,  RBRACK,  UNDEF,  ATEND,  ARG,  ARG2,
C           ARG3,  ARG4,  ALIST,  SUBR0,  SUBR1,  SUBR2,
C           SUBR3,  A2,  A8,  A9,  FUNARG,  EXPR,
C           FEXPR,  SUBR,  FSUBR,  T,  SYS,  RDPNT,
C           PRTPNT,  ATENDA,  BSTACK,  DREG,  CC,  APVAL,
C           ARGSV,  ARITH
C           INTEGER PNAMEP,PNAMEN
C           INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR
C-----LIS00510
C*           COMMON AND INTEGER DECLARATIONS ENDED
C*           -----
C*           C0000 LOCAL DECLARATIONS
C*           DIMENSION ISIFF(8)
C*           LOGICAL*1 IPRI(80)
C*           INTEGER GET,ADDAL,CONS
C*           THE WORD SIZE OF KAR,CAR AND KDR,CDR MUST MATCH
C*           INTEGER*2 KAR(10), KDR(10)

```

2 9
2 10
2 11
2 12
2 13
2 14
2 15
2 16
2 17
2 18
2 19
2 20
2 21
2 22
2 23
2 24
2 25
2 26
2 27
2 28
2 29
2 30
2 31
2 32
2 33
2 34
2 35
2 36
2 37
2 38
2 39
2 40
2 41
2 42
2 43
2 44
2 45
2 46
2 47
2 48
2 49
2 50
3 1
3 2
3 3
3 4
3 5
3 6
3 7
3 8
3 9
3 10
3 11
3 12
3 13
3 14
3 15
3 16
3 17
3 18

```
INTEGER IPN(2)  
EQUIVALENCE (IPN(1),IPN1),(IPN(2),IPN2)  
EQUIVALENCE (KAR(1),CAR(1))  
EQUIVALENCE (KDR(1),CDR(1))  
EQUIVALENCE (NFREE,NFREET)  
EQUIVALENCE (IRES,ARG)  
  
C  
C-----  
C0000 INITIALIZATION  
LERR(1)=2  
LERR(3)=9  
IGENF=0  
IPN1=0  
ND=25  
DO 10 I=1,24  
10 DREG(I)=0  
CALL INIT(1)  
A4=A9+1  
A6=A9+2  
  
C  
C  
4 CALL FPUSH(1,500)  
ARG2=ALIST  
GOTO 1000  
C@@@ ATTEMPTED RETURN FROM TOP-LEVEL LOOP  
500 WRITE (3,501)  
501 FCRMAT (36H DONT USE ( - YOU ARE NOT IN A BREAK)  
  
C  
C0000 RESTART OF INTERPRETER  
2 IP=0  
JP=NSTACK+1  
GOTO 4  
  
C  
C0000 RETURN IN BREAK AND FROM TOP LEVEL LOOP  
997 JP=JP+1  
C0000 RECURSIVE RETURN PROCEDURE  
998 I=STACK(IP)  
IP=IP-1  
IF (IP) 990,999,999  
990 WRITE (3,991)  
991 FORMAT (24H FUNCTION PDL IS EMPTY.?)  
GOTO 2  
999 GOTO (500, 1103, 3000, 3501, 4103, 1000, 4501,  
C0000 I= 1 2 3 4 5 6 7  
* 2, 5101, 6101, 7710, 9710, 9741,  
C0000 I= 8 9 10 11 12 13  
* 9621, 9628, 9653, 9510, 9523, 9533,  
C I= 14 15 16 17 18 19  
* 9667, 9664, 9558, 9554, 9556,  
C I= 20 21 22 23 24  
* 500),I  
  
C  
C-----  
C.... RECURSIVE FUNCTION LOOP ()  
1000 CALL APUSH(ARG2)  
1001 ARG2=STACK(JP)  
CALL FPUSH(2,1103)  
  
C  
INTEGER BLCOM(64),CREATU/0/,LOADU/0/  
EQUIVALENCE (BLCOM,IP)
```

LIS00590
LIS00600
LIS00610
LIS00620
LIS00630
LIS00640
LIS00650
LIS00660
LIS00670
LIS00680
LIS00690
LIS00700
LIS00710
LIS00720
LIS00730
LIS00740
LIS00750
LIS00760
LIS00770
LIS00780
LIS00790
LIS00800
LIS00810
LIS00820
LIS00830
LIS00840
LIS00850
LIS00860
LIS00870
LIS00880
LIS00890
LIS00900
LIS00910
LIS00920
LIS00930
LIS00940
LIS00950
LIS00960
LIS00970
LIS00980
LIS00990
LIS01000
LIS01010
LIS01020
LIS01030
LIS01040
LIS01050
LIS01060
LIS01070
LIS01080
LIS01090
LIS01100
LIS01110
LIS01120
LIS01130
LIS01140
LIS01150
LIS01160
LIS01170
LIS01180

3	19	C		LIS01190
3	20		IF(LOADU.EQ.0) GO TO 1020	LIS01200
3	21		REWIND LOADU	LIS01210
3	22		READ(LOADU)	LIS01220
3	23		* BLCCM,NILCAR,CAR,NILCDR,CDR,PNAME,PNAME1,PNAME2,STACK,NATOMP	LIS01230
3	24		LOADU=0	LIS01240
3	25		GO TO 1090	LIS01250
3	26	C		LIS01260
3	27	1020	IF(CREATU.EQ.0) GO TO 1090	LIS01270
3	28		REWIND CREATU	LIS01280
3	29		WRITE(CREATU)	LIS01290
3	30		* BLCCM,NILCAR,CAR,NILCDR,CDR,PNAME,PNAME1,PNAME2,STACK,NATOMP	LIS01300
3	31		CREATU=0	LIS01310
3	32	1090	CONTINUE	LIS01320
3	33	C		LIS01330
3	34		GOTO (1100,1200,1300),MODE	LIS01340
3	35	C		LIS01350
3	36	C0000	USE EVAL IN TOP LEVEL LOOP	LIS01360
3	37	1100	IARG=IREAD(0)	LIS01370
3	38		IF (DREG(24)) 1150,1150,1151	LIS01380
3	39	1150	CALL IPRINT(IARG)	LIS01390
3	40	1151	CONTINUE	LIS01400
3	41		IF (IARG) 1101,4009,4009	LIS01410
3	42	4009	ARG=IARG	LIS01420
3	43		GOTO 4000	LIS01430
3	44	C0000	IF IREAD RETURNS A NEGATIVE VALUE, IT WANTS US TO MAKE	LIS01440
3	45	C0000	A RETURN FROM LOOP	LIS01450
3	46	1101	IP=IP-1	LIS01460
3	47		JP=JP+1	LIS01470
3	48		GOTO 4000	LIS01480
3	49	C@@@	RETURN POSITION FROM EVAL, EVALQUOTE, AND APPLY SYS	LIS01490
3	50	1103	IF (DREG(24)) 1104,1104,1001	LIS01500
4	1	1104	CALL IPFINT(IRES)	LIS01510
4	2		GOTO 1001	LIS01520
4	3	C		LIS01530
4	4	C0000	USE EVALQUOTE	LIS01540
4	5	1200	IARG=IREAD(0)	LIS01550
4	6		IF (IARG) 1202,1201,1201	LIS01560
4	7	1201	ARG3=IREAD(0)	LIS01570
4	8		IF (DREG(24)) 1152,1152,1153	LIS01580
4	9	1152	CALL IPRINT(IARG)	LIS01590
4	10		CALL IPRINT(ARG3)	LIS01600
4	11	1153	CONTINUE	LIS01610
4	12		ARG=IARG	LIS01620
4	13		IF (ARG3) 1205,2000,2000	LIS01630
4	14	1202	ARG3=IREAD(0)	LIS01640
4	15		IF (DREG(24)) 1154,1154,1155	LIS01650
4	16	1154	CALL IPRINT(IARG)	LIS01660
4	17		CALL IPFINT(ARG3)	LIS01670
4	18	1155	CONTINUE	LIS01680
4	19		IF (ARG3) 1205,1203,1203	LIS01690
4	20	1203	IP=IP-1	LIS01700
4	21		JP=JP+1	LIS01710
4	22		GOTO 2000	LIS01720
4	23	1205	WRITE (3,1206)	LIS01730
4	24	1206	FORMAT (21H BAD INPUT, TRY AGAIN)	LIS01740
4	25		GOTO 1200	LIS01750
4	26	C		LIS01760
4	27	C0000	USE APPLY	LIS01770
4	28	1300	ARG=GET(SYS,EXPR)	LIS01780

4	29	IF(ARG) 1340,1340,1302	LIS01790
4	30	1302 ARG3=NIL	LIS01800
4	31	GOTO 3000	LIS01810
4	32	1340 WRITE (3,1341)	LIS01820
4	33	1341 FORMAT (38H MODE = 3 BUT SYS UNDEF. TALK TO EVAL.)	LIS01830
4	34	GOTO 1100	LIS01840
4	35	C	LIS01850
4	36	C-----	LIS01860
4	37	C.... RECURSIVE FUNCTION EVQT (SIMILAR TO EVALQUOTE)	LIS01870
4	38	2000 IF (GET(ARG,FEXPR)) 2010,2010,2020	LIS01880
4	39	2010 IF (ARG-SUBR) 3000,3000,2011	LIS01890
4	40	2011 IF (ARG-FSUBR) 2020,2020,3000	LIS01900
4	41	2020 ARG=CONS(ARG,ARG3)	LIS01910
4	42	GOTO 4000	LIS01920
4	43	C	LIS01930
4	44	C-----	LIS01940
4	45	C.... RECURSIVE FUNCTION APPYL (SIMILAR TO APPLY)	LIS01950
4	46	3000 IF (ARG) 9000,998,3001	LIS01960
4	47	3001 IF (ARG-NATCM) 3002,3002,3200	LIS01970
4	48	3002 IF (CDR(ARG)) 3101,3110,3101	LIS01980
4	49	C	LIS01990
4	50	C0000 EXPR0S	LIS02000
5	1	3101 L=GET(ARG,EXPR)	LIS02010
5	2	IF (L) 3110, 3110, 3102	LIS02020
5	3	3102 ARGS AV =ARG	LIS02030
5	4	ARG=L	LIS02040
5	5	GOTO 3001	LIS02050
5	6	C0000 THAT TAKES US TO APPLY	LIS02060
5	7	C	LIS02070
5	8	C0000 SUBR0S	LIS02080
5	9	3110 IF (ARG-SUBR) 9001,9001,3130	LIS02090
5	10	C	LIS02100
5	11	C0000 ALIST DEFINED FUNCTIONS	LIS02110
5	12	3130 CALL FPUSH (3,3131)	LIS02120
5	13	ARG4=A2	LIS02130
5	14	C0000 A2 IS SUPPOSED TO BE AN INDICATION OF ERROR A2	LIS02140
5	15	GOTO 7000	LIS02150
5	16	C0000 THAT TAKES US TO CASSOC	LIS02160
5	17	C@@@ RETURN FROM CASSOC	LIS02170
5	18	C3131 = 3000	LIS02180
5	19	C	LIS02190
5	20	C0000 LAMEDA EXPRESSIONS	LIS02200
5	21	3200 IF (CAR(ARG)-LAMBDA) 3300,3201,3300	LIS02210
5	22	3201 ICAR=KAR(CDR(ARG))	LIS02220
5	23	IF (ICAR.LE.0) GO TO 3202	LIS02230
5	24	ARG2=ADDAL(ICAR,ARG3,ARG2,J)	LIS02240
5	25	IF (J.NE.0) GO TO 9922	LIS02250
5	26	3202 ARG=KAR(KDR(CDR(ARG)))	LIS02260
5	27	GOTO 4000	LIS02270
5	28	C0000 THAT TAKES US TO EVAL	LIS02280
5	29	C	LIS02290
5	30	C0000 FUNARG EXPRESSIONS	LIS02300
5	31	3300 IF (CAR(ARG)-FUNARG) 3400,3301,3400	LIS02310
5	32	3301 ARG2=KAR(KDR(CDR(ARG)))	LIS02320
5	33	ARG=KAR(CDR(ARG))	LIS02330
5	34	GOTO 3000	LIS02340
5	35	C	LIS02350
5	36	C0000 LABEL EXPRESSIONS	LIS02360
5	37	3400 IF (CAR(ARG)-LABEL) 3500,3401,3500	LIS02370
5	38	3401 K=CDR(ARG)	LIS02380

5	39	L=CDR(K)	LIS02390
5	40	ARG=CAR(L)	LIS02400
5	41	ICAR=CAR(K)	LIS02410
5	42	ARG2=CONS(CONS(ICAR, ARG), ARG2)	LIS02420
5	43	GOTO 3000	LIS02430
5	44	C	LIS02440
5	45	C0000 OTHER EXPRESSIONS (TO BE EVALUATED)	LIS02450
5	46	3500 CALL APUSH (ARG2)	LIS02460
5	47	CALL APUSH(ARG3)	LIS02470
5	48	CALL FPUSH (4,3501)	LIS02480
5	49	GOTO 4000	LIS02490
5	50	C@@@@ RETURN FROM EVAL	LIS02500
6	1	3501 CALL APCP(ARG3)	LIS02510
6	2	CALL APCP(ARG2)	LIS02520
6	3	GOTO 3000	LIS02530
6	4	C-----	LIS02540
6	5	C... RECURSIVE FUNCTION EVAL	LIS02550
6	6	4000 IF (ARG-NATOM) 4001,4001,4090	LIS02560
6	7	C	LIS02570
6	8	C0000 ARG AN ATCM	LIS02580
6	9	4001 IF (ARG) 9922,998,4002	LIS02590
6	10	4002 I=GET(ARG,APVAL)	LIS02600
6	11	IF (I) 7002,4003,4003	LIS02610
6	12	4003 IRES=I	LIS02620
6	13	GOTO 998	LIS02630
6	14	C0000 7002 = CASSOC(-,-,A8)	LIS02640
6	15	C	LIS02650
6	16	4090 IF (ARG-NFREE) 4100,4100,998	LIS02660
6	17	C0000 ARG NOT AN ATCM	LIS02670
6	18	4100 IF (CAR(ARG)-NATOM) 4108,4108,4101	LIS02680
6	19	4101 IF (CAR(ARG)-NFREE) 4600,4600,4500	LIS02690
6	20	4108 ARGSAV = CAR(ARG)	LIS02700
6	21	IF(KDR(CAR(ARG))) 4102,4300,4102	LIS02710
6	22	C	LIS02720
6	23	C0000 EXPR IN CAR(ARG)	LIS02730
6	24	4102 ICAR=CAR(ARG)	LIS02740
6	25	L=GET(ICAR,EXPR)	LIS02750
6	26	IF (L) 4200,4200,4104	LIS02760
6	27	4105 L=-L	LIS02770
6	28	4104 CALL APUSH(ARG2)	LIS02780
6	29	CALL APUSH(L)	LIS02790
6	30	CALL FPUSH(5,4103)	LIS02800
6	31	ARG=CDR(ARG)	LIS02810
6	32	GOTO 5000	LIS02820
6	33	C@@@@ RETURN FROM EVLIS	LIS02830
6	34	4103 ARG3=IRES	LIS02840
6	35	CALL APCP(ARG)	LIS02850
6	36	CALL APCP(ARG2)	LIS02860
6	37	GOTO 3000	LIS02870
6	38	C0000 TAKES US TO CHOICE OF APPLY OR MACH	LIS02880
6	39	C	LIS02890
6	40	C0000 FEXPR	LIS02900
6	41	4200 ICAR=CAR(ARG)	LIS02910
6	42	L=GET(ICAR,FEXPR)	LIS02920
6	43	IF (L) 4300,4300,4201	LIS02930
6	44	4201 ICDR=CDR(ARG)	LIS02940
6	45	ARG3=CONS(ICDR,CONS(ARG2,NIL))	LIS02950
6	46	ARG=L	LIS02960
6	47	GOTO 3001	LIS02970
6	48	C	LIS02980

6	49	C0000	SUBR, FSUBR	LIS02990
6	50	4300	L=CAR(ARG)	LIS03000
7	1		IF (L-SUBR) 4105,4105,4400	LIS03010
7	2	4400	IF (L-FSUBR) 4401,4401,4500	LIS03020
7	3	4401	ARG=CDR(ARG)	LIS03030
7	4		GOTO 9591	LIS03040
7	5	C0000		LIS03050
7	6	C0000	FUNCTION DEFINED ON A-LIST	LIS03060
7	7	4500	ICDR=CDR(ARG)	LIS03070
7	8		CALL APUSH(ICDR)	LIS03080
7	9		CALL FPUSH(7,4501)	LIS03090
7	10		ARG=CAR(ARG)	LIS03100
7	11		ARG4=A9	LIS03110
7	12		GOTO 7000	LIS03120
7	13	C@@@	RETURN FROM CASSOC	LIS03130
7	14	4501	CALL APOP(I)	LIS03140
7	15		ARG=CONS(IRES,I)	LIS03150
7	16		GOTO 4000	LIS03160
7	17	C0000		LIS03170
7	18	C0000	EVALUATE THE FUNCTION	LIS03180
7	19	4600	L=CAR(ARG)	LIS03190
7	20		GOTO 4104	LIS03200
7	21	C	-----	LIS03210
7	22	C....	RECURSIVE FUNCTION EVLIS	LIS03220
7	23	5000	IF(ARG) 9933,998,5010	LIS03230
7	24	5010	CALL APUSH(CONS(NIL,NIL))	LIS03240
7	25		CALL APUSH(NIL)	LIS03250
7	26		STACK(JP)=STACK(JP+1)	LIS03260
7	27		CALL APUSH(ARG)	LIS03270
7	28		CALL APUSH(ARG2)	LIS03280
7	29	C0000	STACK(JP+3) = BEGINNING OF RESULT LIST FROM EVLIS	LIS03290
7	30	C0000	STACK(JP+2) = END OF RESULT LIST	LIS03300
7	31	5020	ARG=CAR(ARG)	LIS03310
7	32		CALL FPUSH(9,5101)	LIS03320
7	33		GOTO 4000	LIS03330
7	34	C@@@	RETURN FROM EVAL	LIS03340
7	35	5101	I=STACK(JP+2)	LIS03350
7	36		CAR(I)=IRES	LIS03360
7	37		ARG=KDR(STACK(JP+1))	LIS03370
7	38		IF (ARG) 5299,5200,5110	LIS03380
7	39	5110	ARG2=STACK(JP)	LIS03390
7	40		STACK(JP+1)=ARG	LIS03400
7	41		CDR(I)=CONS(NIL,NIL)	LIS03410
7	42		STACK(JP+2)=CDR(I)	LIS03420
7	43		GOTO 5020	LIS03430
7	44	5200	IRES=STACK(JP+3)	LIS03440
7	45		K=STACK(JP+2)	LIS03450
7	46		CDR(K)=NIL	LIS03460
7	47		JP=JP+4	LIS03470
7	48		GOTO 998	LIS03480
7	49	5299	JP=JP+4	LIS03490
7	50		GOTO 9933	LIS03500
8	1	C	-----	LIS03510
8	2	C....	RECURSIVE FUNCTION EVCON	LIS03520
8	3	6000	IF (ARG) 9933,998,6010	LIS03530
8	4	6010	CALL APUSH(ARG2)	LIS03540
8	5		CALL APUSH(ARG)	LIS03550
8	6		ICAR=CAR(ARG)	LIS03560
8	7	6020	IF (ICAR.LT.0.OR.ICAR.GT.NFREE) GO TO 9933	LIS03570
8	8			LIS03580

8	9	ARG=KAR(ICAR)	LIS03590
8	10	CALL FPUSH(10,6101)	LIS03600
8	11	GOTO 4000	LIS03610
8	12	C@@@@ RETURN FROM EVAL	LIS03620
8	13	6101 ARG2=STACK(JP+1)	LIS03630
8	14	IF (IRES) 6200,6110,6200	LIS03640
8	15	6110 ARG=KDR(STACK(JP))	LIS03650
8	16	IF (ARG) 9933,6201,6112	LIS03660
8	17	6112 STACK(JP)=ARG	LIS03670
8	18	GOTO 6020	LIS03680
8	19	6200 ICAR=KDR(KAR(STACK(JP)))	LIS03690
8	20	IF (ICAR.LT.0.OR.ICAR.GT.NFREE) GO TO 9933	LIS03700
8	21	ARG=KAR(ICAR)	LIS03710
8	22	JP=JP+2	LIS03720
8	23	GOTO 4000	LIS03730
8	24	6201 JP=JP+2	LIS03740
8	25	GOTO 998	LIS03750
8	26	C	LIS03760
8	27	C-----	LIS03770
8	28	C.... SEMI-RECURSIVE FUNCTION CASSOC, SASSOC	LIS03780
8	29	C0000 ARGUMENTS IN ARG, ARG2, ARG4, ARG2 AND ARG3 ARE NOT	LIS03790
8	30	C0000 TO BE DESTROYED.	LIS03800
8	31	C0000 CASSOC(-,-,-) = CDR(SASSOC(-,-,-))	LIS03810
8	32	C0000 ENTRY SASSOC	LIS03820
8	33	7010 IASSOC=0	LIS03830
8	34	GOTO 7020	LIS03840
8	35	C0000 ENTRY CASSOC(-,-,A8)	LIS03850
8	36	7002 ARG4=A8	LIS03860
8	37	C0000 ENTRY CASSOC	LIS03870
8	38	7000 IASSOC=1	LIS03880
8	39	7020 K=ARG2	LIS03890
8	40	7030 IF (K) 9933,7700,7040	LIS03900
8	41	7040 ICAR=CAR(K)	LIS03910
8	42	IF (ICAR.LT.0.OR.ICAR.GT.NFREE) GO TO 9933	LIS03920
8	43	IF (ARG-KAR(ICAR)) 7050,7100,7050	LIS03930
8	44	7050 K=CDR(K)	LIS03940
8	45	GOTO 7030	LIS03950
8	46	C0000 ORDINARY RETURN	LIS03960
8	47	7100 IF (IASSOC) 7101,7101,7102	LIS03970
8	48	7101 IRES=CAR(K)	LIS03980
8	49	GOTO 998	LIS03990
8	50	7102 IRES=KDR(CAR(K))	LIS04000
9	1	GOTO 998	LIS04010
9	2	C0000 APPLY ERROR FUNCTION (THIRD ARGUMENT)	LIS04020
9	3	7700 CALL APUSH(ARG2)	LIS04030
9	4	CALL APUSH(ARG3)	LIS04040
9	5	CALL FPUSH(11,7710)	LIS04050
9	6	ARG3=CONS(ARG,CONS(ARG2,NIL))	LIS04060
9	7	ARG=ARG4	LIS04070
9	8	GOTO 3000	LIS04080
9	9	C@@@@ RETURN FROM APPYL	LIS04090
9	10	7710 CALL APCP(ARG3)	LIS04100
9	11	CALL APCP(ARG2)	LIS04110
9	12	GOTO 998	LIS04120
9	13	C	LIS04130
9	14	C-----	LIS04140
9	15	C.... RECURSIVE FUNCTION MACH (LIKE APPYL, BUT FOR SUBRS	LIS04150
9	16	C0000 AND FSUBROS)	LIS04160
9	17	C	LIS04170
9	18	C0000 ON ENTRY HERE, THE FIRST ARGUMENT TO MACH IS NEGATIVE	LIS04180

9	19	C0000 AND SUPPOSED TO BE A SUBR	LIS04190
9	20	9000 L=-ARG	LIS04200
9	21	GOTO 9002	LIS04210
9	22	C0000 ANOTHER ENTRY, ARG POSITIVE	LIS04220
9	23	9001 L=ARG	LIS04230
9	24	9002 ARGSV = L	LIS04240
9	25	IF (L-SUBR2) 9003,9003,9300	LIS04250
9	26	9003 IF (L-SUBR1) 9005,9005,9200	LIS04260
9	27	9005 IF(L-SUBR0) 9006,9006,9004	LIS04270
9	28	C	LIS04280
9	29	C0000 RETURN CHANNELS FOR TRUTH-VALUED FUNCTIONS	LIS04290
9	30	9010 IRES=NIL	LIS04300
9	31	GOTO 998	LIS04310
9	32	9011 IRES=T	LIS04320
9	33	GOTO 998	LIS04330
9	34	C	LIS04340
9	35	C.....	LIS04350
9	36	C0000 SUBR00S	LIS04360
9	37	9006 GO TO (LIS04370
9	38	* 9031, 9032, 9033, 9035,	LIS04380
9	39	C ADVAN BREAK EJECT FORCEGBC	LIS04390
9	40	* 9038, 2, 9040, 9041, 9042, 9043,	LIS04400
9	41	C READ RESTART SILEN TALK TERPRI EXIT	LIS04410
9	42	* 92011), L	LIS04420
9	43	C	LIS04430
9	44	C0000 SUBR10S	LIS04440
9	45	9004 IF (ARG3.LE.NATOM.OR.ARG3.GT.NFREE) GO TO 9922	LIS04450
9	46	ARG=CAR(ARG3)	LIS04460
9	47	IFLAG=0	LIS04470
9	48	IF (ARG.GT.NFREE) IFLAG=1	LIS04480
9	49	L = L-SUBRO	LIS04490
9	50	GOTO (LIS04500
10	1	* 9101, 9102, 9103, 9104, 9105, 9106, 9107,	LIS04510
10	2	C ADD1 ATOM CAR CDR CAAR CADR CDAR	LIS04520
10	3	* 9108, 9109, 9110, 9111, 9112, 9113, 9114,	LIS04530
10	4	C CDDR CLEARB GENS INUNT NULL NUMBP OUTUN	LIS04540
10	5	* 9115, 9116, 9117, 9118,91185, 9119,	LIS04550
10	6	C PACKL PRIN1 PRINT PRPOS RDPOS SETBIT	LIS04560
10	7	* 9120, 9121, 9122, 9123, 9740, 9780, 9112,	LIS04570
10	8	C SUB1 TESTB UNPACK ZEROP GOTO RETURN NOT	LIS04580
10	9	* 9034, 9037,	LIS04590
10	10	C MODE OBLIST	LIS04600
10	11	* 9124, 9125,	LIS04610
10	12	C CREATE LOAD	LIS04620
10	13	* 92011), L	LIS04630
10	14	C	LIS04640
10	15	C	LIS04650
10	16	C9031 ADVANCE	LIS04660
10	17	9031 IRES=MATOM(CC,1)	LIS04670
10	18	CALL SHIFT(CC)	LIS04680
10	19	GO TO 998	LIS04690
10	20	C	LIS04700
10	21	C9032 BREAK	LIS04710
10	22	9032 WRITE (3,9932)	LIS04720
10	23	9932 FCRMAT (24H BREAK CALLED BY PROGRAM)	LIS04730
10	24	GOTO 1000	LIS04740
10	25	C	LIS04750
10	26	C9033 EJECT	LIS04760
10	27	9033 WRITE (3,9098)	LIS04770
10	28	9098 FORMAT (1H1)	LIS04780

10	29	IRES = -1	LIS04790
10	30	GOTO 998	LIS04800
10	31	C	LIS04810
10	32	C9034 MODE	LIS04820
10	33	9034 IF (IFLAG.LE.0) GO TO 9933	LIS04830
10	34	K=MODE	LIS04840
10	35	MODE=ARG-NUMADD	LIS04850
10	36	IRES = K + NUMADD	LIS04860
10	37	GOTO 998	LIS04870
10	38	C	LIS04880
10	39	C	LIS04890
10	40	C9035 FORCEGEC	LIS04900
10	41	9035 CDR(NFREEP)=NIL	LIS04910
10	42	IDUM=CONS(NIL,NIL)	LIS04920
10	43	IRES=LISTNB(NFREEP)+NUMADD	LIS04930
10	44	GOTO 998	LIS04940
10	45	C	LIS04950
10	46	C9037 OBLIST	LIS04960
10	47	9037 I=ARG	LIS04970
10	48	IDUM=CONS(I,NIL)	LIS04980
10	49	IRES=IDUM	LIS04990
10	50	90370 IF(I-NATOMP) 90371,998,90371	LIS05000
11	1	90371 I=I+1	LIS05010
11	2	CDR(IDUM)=CCNS(I,NIL)	LIS05020
11	3	IDUM=CDR(IDUM)	LIS05030
11	4	GO TO 90370	LIS05040
11	5	C	LIS05050
11	6	C9038 READ	LIS05060
11	7	9038 IRES=IREAD(0)	LIS05070
11	8	IF (IRES) 9604,998,998	LIS05080
11	9	9604 WRITE (3,9608)	LIS05090
11	10	9608 FORMAT (16H (INPUT TO READ /6H BREAK)	LIS05100
11	11	GOTO 1000	LIS05110
11	12	C	LIS05120
11	13	C9040 SILENCE	LIS05130
11	14	9040 DREG(24)=T	LIS05140
11	15	IRES = -1	LIS05150
11	16	GOTO 998	LIS05160
11	17	C	LIS05170
11	18	C9041 TALK	LIS05180
11	19	9041 DREG(24)=NIL	LIS05190
11	20	IRES = -1	LIS05200
11	21	GOTO 998	LIS05210
11	22	C	LIS05220
11	23	C9042 TERPRI	LIS05230
11	24	9042 CALL PRINAT(-1,IDUM)	LIS05240
11	25	IRES = -1	LIS05250
11	26	GOTO 998	LIS05260
11	27	C	LIS05270
11	28	C9043 EXIT	LIS05280
11	29	9043 CALL EXIT	LIS05290
11	30	C	LIS05300
11	31	C9101 ADD1	LIS05310
11	32	9101 IRES = ARITH(1,ARG,NUMADD+1,ARGSAV)	LIS05320
11	33	GOTO 998	LIS05330
11	34	C	LIS05340
11	35	C9102 ATOM	LIS05350
11	36	9102 IF (ARG) 9010,9601,9601	LIS05360
11	37	9601 IF (ARG-NATOM) 9011,9011,9113	LIS05370
11	38	C	LIS05380

```

11 39 C9103 CAR
11 40 9103 IGO = 1
11 41 91030 IF (ARG.LE.NATOM.OR.ARG.GT.NFREE) GO TO 9966
11 42 IRES = CAR(ARG)
11 43 GO TO (998, 9103, 9104) , IGO
11 44 C
11 45 C9104 CDR
11 46 9104 IRES=CDR(ARG)
11 47 GOTO 998
11 48 C
11 49 C9105 CAAR
11 50 9105 IGO = 2
12 1 GO TO 91030
12 2 C
12 3 C9106 CADR
12 4 9106 IRES=CDR(ARG)
12 5 GO TO 9103
12 6 C
12 7 C9107 CDAR
12 8 9107 IGO = 3
12 9 GO TO 91030
12 10 C
12 11 C9108 CDDR
12 12 9108 IRES=CDR(ARG)
12 13 GO TO 9104
12 14 C
12 15 C9109 CLEARBIT
12 16 9109 IF (IFLAG.LE.0) GO TO 9933
12 17 I=ARG-NUMADD
12 18 IF (I * (ND - 1)) 9611,9611,9610
12 19 9610 DREG(I)=NIL
12 20 IRES=NIL
12 21 GOTO 998
12 22 9611 WRITE (3,9612) I
12 23 9612 FORMAT (38H BIT ARRAY USED OUT OF BOUNDS, INDEX = , I4)
12 24 GOTO 1000
12 25 C
12 26 C9110 GENSYM
12 27 9110 IF(IGENF-99) 91101,91102,91101
12 28 C THIS IS THE INITIAL POINT FOR GENSYM
12 29 91101 INDSIF=0
12 30 IGENF=99
12 31 91102 ISIF=INDSIF
12 32 DO 91103 JENIND=1,4
12 33 JJ=ISIF-ISIF/10*10
12 34 ISIFF(JENIND)=IFIG(JJ+1)
12 35 91103 ISIF=ISIF/10
12 36 DO 91104 JENIND=1,4
12 37 III=5-JENIND
12 38 91104 CALL PUTCH(IPN2,ISIFF(III),JENIND)
12 39 CALL MVC(IPN1,'0000',4,IPN1,PNAME(PNAMEP(ARG)),MIN0(PNAME2,4))
12 40 INDSIF=INDSIF+1
12 41 IRES=MATOM(IPN1,8)
12 42 C L PCINTS HERE TO THE ATOM GENSYM
12 43 GO TO 998
12 44 C
12 45 C9111 INUNIT
12 46 9111 IF (IFLAG.LE.0) GO TO 9933
12 47 K=LUNIN
12 48 LUNIN=ARG-NUMADD

```

```

LIS05390
LIS05400
LIS05410
LIS05420
LIS05430
LIS05440
LIS05450
LIS05460
LIS05470
LIS05480
LIS05490
LIS05500
LIS05510
LIS05520
LIS05530
LIS05540
LIS05550
LIS05560
LIS05570
LIS05580
LIS05590
LIS05600
LIS05610
LIS05620
LIS05630
LIS05640
LIS05650
LIS05660
LIS05670
LIS05680
LIS05690
LIS05700
LIS05710
LIS05720
LIS05730
LIS05740
LIS05750
LIS05760
LIS05770
LIS05780
LIS05790
LIS05800
LIS05810
LIS05820
LIS05830
LIS05840
LIS05850
LIS05860
LIS05870
LIS05880
LIS05890
LIS05900
LIS05910
LIS05920
LIS05930
LIS05940
LIS05950
LIS05960
LIS05970
LIS05980

```

12	49	CALL SHIFT(123456)	LIS05990
12	50	IRES=K+NUMADD	LIS06000
13	1	GOTO 998	LIS06010
13	2	C	LIS06020
13	3	C9112 NULL AND NOT	LIS06030
13	4	9112 IF (ARG) 9010, 9011, 9010	LIS06040
13	5	C	LIS06050
13	6	C9113 NUMBERP	LIS06060
13	7	9113 IF (ARG-NFREE) 9010,9010,9011	LIS06070
13	8	C	LIS06080
13	9	C9114 OUTUNIT	LIS06090
13	10	9114 IF (IFLAG.LE.0) GO TO 9933	LIS06100
13	11	K=LUNUT	LIS06110
13	12	LUNUT=ARG-NUMADD	LIS06120
13	13	IRES=K+NUMADD	LIS06130
13	14	GOTO 998	LIS06140
13	15	C	LIS06150
13	16	C9115 PACKLIST	LIS06160
13	17	9115 IDUM=1	LIS06170
13	18	ISUM=0	LIS06180
13	19	I=ARG	LIS06190
13	20	III=0	LIS06200
13	21	91151 II=CAR(I)	LIS06210
13	22	IF(NFREE-II) 91153,91154,91154	LIS06220
13	23	91153 II=II-NUMADD	LIS06230
13	24	II=II-II/10*10	LIS06240
13	25	ISUM=10*ISUM+II	LIS06250
13	26	CALL SBYT(IFIG(II+1),IPRI,III)	LIS06260
13	27	GO TO 91155	LIS06270
13	28	91154 CALL MVC(IPRI(III+1),PNAME(PNAMEP(II)),PNAME2)	LIS06280
13	29	III=III+PNAME2	LIS06290
13	30	IDUM=2	LIS06300
13	31	91155 I=CDR(I)	LIS06310
13	32	IF(I) 91151,91152,91151	LIS06320
13	33	91152 IRES=MATOM(IPRI,III)	LIS06330
13	34	IF(IDUM-1) 998,91156,998	LIS06340
13	35	91156 IRES=ISUM+NUMADD	LIS06350
13	36	GO TO 998	LIS06360
13	37	C	LIS06370
13	38	C9116 PRINI	LIS06380
13	39	9116 I=ARG	LIS06390
13	40	CALL PRINAT(4,I)	LIS06400
13	41	IF(P RTPNT .GT. 1) RTPNT=P RTPNT-1	LIS06410
13	42	GCTO 998	LIS06420
13	43	C	LIS06430
13	44	C9117 PRINT	LIS06440
13	45	9117 CALL IPRINT(ARG)	LIS06450
13	46	GOTO 998	LIS06460
13	47	C9118 PRINTPOS	LIS06470
13	48	9118 IF (ARG) 91182,91181,91182	LIS06480
13	49	91181 IRES=P RTPNT+NUMADD-1	LIS06490
13	50	GO TC 998	LIS06500
14	1	91182 IF (IFLAG.LE.0) GO TO 9933	LIS06510
14	2	IF (ARG-NUMADD) 91189,91189,91184	LIS06520
14	3	91184 RTPNT=ARG-NUMADD	LIS06530
14	4	GO TO 998	LIS06540
14	5	C	LIS06550
14	6	C91185 READPCS	LIS06560
14	7	91185 IF (ARG) 91187,91186,91187	LIS06570
14	8	91186 IRES=RDPNT+NUMADD-1	LIS06580

14	9								LIS06590
14	10								LIS06600
14	11	91187	IF (IFLAG.LE.0)	GO TO 9933					LIS06610
14	12								LIS06620
14	13	91188	RDPNT=ARG-NUMADD						LIS06630
14	14								LIS06640
14	15	91189	WRITE(3,96189)						LIS06650
14	16	96189	FORMAT(45H ARGUMENT FOR READPOS OR PRINTPOS LESS THAN 0						LIS06660
14	17		+ /6H BREAK)						LIS06670
14	18		GO TO 1000						LIS06680
14	19	C							LIS06690
14	20	C9119	SETBIT						LIS06700
14	21	9119	IF (IFLAG.LE.0)	GO TO 9933					LIS06710
14	22		I=ARG-NUMADD						LIS06720
14	23		IF (I.EQ.100)CALL SVC(15)						LIS06730
14	24		IF (I*(ND-I))	9611,9611,9613					LIS06740
14	25	9613	DREG(I)=T						LIS06750
14	26		IRES=NIL						LIS06760
14	27		GOTO 998						LIS06770
14	28	C							LIS06780
14	29	C9120	SUB1						LIS06790
14	30	9120	IRES = ARITH(2,ARG,NUMADD+1,ARGSAV)						LIS06800
14	31		GOTO 998						LIS06810
14	32	C							LIS06820
14	33	C9121	TESTBIT						LIS06830
14	34	9121	IF (IFLAG.LE.0)	GO TO 9933					LIS06840
14	35		I=ARG-NUMADD						LIS06850
14	36		IF (I*(ND-I))	9611,9611,9614					LIS06860
14	37	9614	ARG=DREG(I)						LIS06870
14	38		GOTO 998						LIS06880
14	39	C							LIS06890
14	40	C9122	UNPACK						LIS06900
14	41	9122	IF(NFREE-ARG)	91225,91220,91220					LIS06910
14	42	91220	II=FNAMEP(ARG)						LIS06920
14	43		JUN=FNAME2-1						LIS06930
14	44		I=MATCH(PNAME(II),1)						LIS06940
14	45		IDUM=CONS(I,NIL)						LIS06950
14	46		IRES=IDUM						LIS06960
14	47		IF(JUN.LE.0)	GO TO 998					LIS06970
14	48		DO 91223 I=1, JUN						LIS06980
14	49		CDR(IDUM)=CONS(MATCH(PNAME(II+I),1),NIL)						LIS06990
14	50	91223	IDUM=CDR(IDUM)						LIS07000
15	1		GO TO 998						LIS07010
15	2	91225	II=ARG-NUMADD						LIS07020
15	3		IRES=NIL						LIS07030
15	4	91226	IRES=CONS(NIL,IRES)						LIS07040
15	5		III=II						LIS07050
15	6		II=II/10						LIS07060
15	7		III=III-10*II+NUMADD						LIS07070
15	8		CAR(IRES)=III						LIS07080
15	9		IF(II)	91226,998,91226					LIS07090
15	10	C							LIS07100
15	11	C9124	CREATE						LIS07110
15	12	9124	IF(ARG.LE.NFREE)	GO TO 9933					LIS07120
15	13		CREATU=ARG-NUMADD						LIS07130
15	14		GO TO 998						LIS07140
15	15	C							LIS07150
15	16	C9125	LOAD						LIS07160
15	17	9125	IF(ARG.LE.NFREE)	GO TO 9933					LIS07170
15	18		LOADU=ARG-NUMADD						LIS07180
			GO TO 998						

15 19
15 20
15 21
15 22
15 23
15 24
15 25
15 26
15 27
15 28
15 29
15 30
15 31
15 32
15 33
15 34
15 35
15 36
15 37
15 38
15 39
15 40
15 41
15 42
15 43
15 44
15 45
15 46
15 47
15 48
15 49
15 50
16 1
16 2
16 3
16 4
16 5
16 6
16 7
16 8
16 9
16 10
16 11
16 12
16 13
16 14
16 15
16 16
16 17
16 18
16 19
16 20
16 21
16 22
16 23
16 24
16 25
16 26
16 27
16 28

```
C
C9123 ZEROP
  9123 IF ( ARG - NUMADD ) 9010,9011,9010
C
C.....
C0000 SUBR20S
  9200 IF (ARG3.LE.NATOM.OR.ARG3.GT.NFREE) GO TO 9922
      ARG=CAR(ARG3)
      ICDR = CDR(ARG3)
      IF (ICDR.LE.NATOM.OR.ICDR.GT.NFREE) GO TO 9922
      ARG3= CAR(ICDR)
C0000 WE USE ARG3 FOR SECOND ARGUMENT OF THE GIVEN FUNCTION.
C0000 BECAUSE THE SUBR2 SET NEEDS THE OLD ARG2 (= ALIST)
      L=L-SUBR1
      GOTO (
*      9201, 9202, 9201, 9219, 9220,
C      A2 A8 A9 A4 A6
*      9203, 9204, 9205, 9206, 9207, 9208, 9209,
C      CONS DIFFR EQ EQUAL EVAL EVL IS GET
*      9210, 9211, 9212, 9213, 9214, 9215, 9216,
C      GREAT LESSP MEMBER NCONC PAIR QUOT RPLACA
*      9217, 9218, 9221, 9660, 9680,
C      RPLACD SET MEMB APPEND DEFOUT
*      92011), L
C
C9201 A2, A9
  9201 L=LERR(L)
      IF ( ARG-NFREE ) 92011,92012,92012
92011 CALL ERROR(1,ARG,&2)
92012 LL=ARG-NUMADD
      WRITE(3,98011) LL
98011 FORMAT (' UNDEFINED FUNCTION ',I6/' RESTART')
      GO TO 2
C
C9202 A8
  9202 CALL ERROR(2,ARG,&2)
C
C9203 CONS
  9203 IRES=CONS(ARG,ARG3)
      GOTO 998
C
C9204 DIFFEREN(CE)
  9204 IRES = ARITH(2,ARG,ARG3,ARGSAV)
      GO TO 998
C
C9205 EQ
  9205 IF ( ARG-ARG3 ) 9010,9011,9010
C
C9219 A4
  9219 CALL ERROR(3,ARG,&1000)
C
C9220 A6
  9220 CALL ERROR(4,ARG,&1000)
C
C9206 EQUAL
  9206 IPE=IF
      JPE=JP
C0000 AUXILIARY RECURSIVE FUNCTION EQUAL1
  9615 IF ( ARG-ARG3 ) 9616,9011,9616
  9616 IF ( ARG-NFREE ) 96161,9617,9617
```

LIS07190
LIS07200
LIS07210
LIS07220
LIS07230
LIS07240
LIS07250
LIS07260
LIS07270
LIS07280
LIS07290
LIS07300
LIS07310
LIS07320
LIS07330
LIS07340
LIS07350
LIS07360
LIS07370
LIS07380
LIS07390
LIS07400
LIS07410
LIS07420
LIS07430
LIS07440
LIS07450
LIS07460
LIS07470
LIS07480
LIS07490
LIS07500
LIS07510
LIS07520
LIS07530
LIS07540
LIS07550
LIS07560
LIS07570
LIS07580
LIS07590
LIS07600
LIS07610
LIS07620
LIS07630
LIS07640
LIS07650
LIS07660
LIS07670
LIS07680
LIS07690
LIS07700
LIS07710
LIS07720
LIS07730
LIS07740
LIS07750
LIS07760
LIS07770
LIS07780

16	29	96161	IF ((ARG-NATOM)*(NFREE+1-ARG)) 9617,9617,9619	LIS07790
16	30	C0000	EQUALITY FAILS	LIS07800
16	31	9617	IRES=NIL	LISC7810
16	32		IP=IPE	LIS07820
16	33		JP=JPE	LIS07830
16	34		GOTC 998	LIS07840
16	35	C0000	ARG NCN-ATOMIC	LIS07850
16	36	9619	IF ((ARG3-NATOM)*(NFREE+1-ARG3)) 9617,9617,9620	LIS07860
16	37	9620	ICDR=CDR(ARG)	LIS07870
16	38		CALL APUSH(ICDR)	LIS07880
16	39		ARG=CAR(ARG)	LIS07890
16	40		ICDR=CDR(ARG3)	LIS07900
16	41		CALL APUSH(ICDR)	LIS07910
16	42		ARG3=CAR(ARG3)	LIS07920
16	43		CALL FPUSH(14,9621)	LISC7930
16	44		GOTO 9615	LIS07940
16	45	C@@@@	RETURN FROM EQUAL1	LIS07950
16	46	9621	CALL APOP(ARG3)	LIS07960
16	47		CALL APOP(ARG)	LIS07970
16	48		GOTO 9615	LIS07980
16	49	C		LIS07990
16	50	C9207	EVAL	LISC8000
17	1	9207	ARG2=ARG3	LISC8010
17	2		GOTO 4000	LISC8020
17	3	C		LISC8030
17	4	C9208	EVLIS	LISC8040
17	5	9208	ARG2=ARG3	LISC8050
17	6		GOTO 5000	LISC8060
17	7	C		LISC8070
17	8	C9209	GET	LISC8080
17	9	9209	IRES=GET(ARG,ARG3)	LISC8090
17	10		IF(IRES.LT.0) IRES=0	LISC8100
17	11		GOTO 998	LISC8110
17	12	C		LISC8120
17	13	C9210	GREATERP	LISC8130
17	14	9210	IF (ARG-ARG3) 9010,9010,9011	LISC8140
17	15	C		LISC8150
17	16	C9211	LESSP	LISC8160
17	17	9211	IF (ARG-ARG3) 9011,9010,9010	LISC8170
17	18	C		LISC8180
17	19	C9221	MEMB	LISC8190
17	20	9221	IRES=MEMB(ARG,ARG3)	LISC8200
17	21		GOTO 998	LISC8210
17	22	C		LISC8220
17	23	C9212	MEMBER	LISC8230
17	24	9212	CALL APUSH(ARG)	LISC8240
17	25		CALL APUSH(ARG3)	LISC8250
17	26	96251	IF(ARG3-NFREE) 9625,9625,9933	LISC8260
17	27	9625	IF (ARG3) 9626,9626,9627	LISC8270
17	28	9626	IRES=NIL	LISC8280
17	29		JP=JP+2	LISC8290
17	30		GOTO 998	LISC8300
17	31	9627	CALL FPUSH(15,9628)	LISC8310
17	32		ARG3=CAR(ARG3)	LISC8320
17	33		GOTO 9206	LISC8330
17	34	C@@@@	RETURN FROM EQUAL	LISC8340
17	35	9628	IF (IRES) 9630,9629,9630	LISC8350
17	36	9629	ARG3=KDR(STACK(JP))	LISC8360
17	37		STACK(JP)=ARG3	LISC8370
17	38		ARG=STACK(JP+1)	LISC8380

17	39		GO TO 96251	LISC8390
17	40	9630	ARG=STACK(JP)	LISC8400
17	41		JP=JP+2	LISC8410
17	42		GOTO 998	LISC8420
17	43	C		LISC8430
17	44	C9213	NCONC	LISC8440
17	45	9213	IF (ARG) 9638,9638,9639	LISC8450
17	46	9638	IRES=ARG3	LISC8460
17	47		GOTO 998	LISC8470
17	48	9639	I=ARG	LISC8480
17	49	9640	IF (CDR(I)) 9641,9641,9642	LISC8490
17	50	9641	CDR(I)=ARG3	LISC8500
18	1		GOTO 998	LISC8510
18	2	9642	I=CDR(I)	LISC8520
18	3		GOTO 9640	LISC8530
18	4	C		LISC8540
18	5	C9214	PAIR	LISC8550
18	6	9214	IRES=ADDAL(ARG,ARG3,NEL,J)	LISC8560
18	7		IF (J.NE.0) WRITE(3,92142)	LISC8570
18	8	92142	FORMAT(* FUNCTION PAIR???? - UNEQUAL LIST LENGTHS*)	LISC8580
18	9		GOTO 998	LISC8590
18	10	C		LISC8600
18	11	C9215	QUOTIENT	LISC8610
18	12	9215	IRES = ARITH(4,ARG,ARG3,ARGSAV)	LISC8620
18	13		GO TO 998	LISC8630
18	14	C		LISC8640
18	15	C9216	RPLACA	LISC8650
18	16	9216	CAR(ARG)=ARG3	LISC8660
18	17		GOTO 998	LISC8670
18	18	C		LISC8680
18	19	C9217	RPLACD	LISC8690
18	20	9217	CDR(ARG)=ARG3	LISC8700
18	21		GOTO 998	LISC8710
18	22	C		LISC8720
18	23	C9218	SET	LISC8730
18	24	9218	ARG4=A4	LISC8740
18	25		CALL FPUSH(16,9653)	LISC8750
18	26		GOTO 7010	LISC8760
18	27	C@@@	RETURN FROM SASSOC	LISC8770
18	28	9653	CDR(ARG)=ARG3	LISC8780
18	29		IRES=ARG3	LISC8790
18	30		GOTO 998	LISC8800
18	31	C9600	APPEND	LISC8810
18	32	9660	IF (ARG.EQ.0)GO TO 9669	LISC8820
18	33		IF (ARG.LE.NATCM.OR.ARG.GT.NFREE) GO TO 9933	LISC8830
18	34		IF (ARG3.EQ.0) GO TO 9661	LISC8840
18	35		IF(ARG3.LE.NATCM.OR.ARG3.GT.NFREE) GO TO 9933	LISC8850
18	36	9661	LASTR=ARG3	LISC8860
18	37		CALL FPUSH(20,9667)	LISC8870
18	38	9662	ICAR= CAR(ARG)	LISC8880
18	39		CALL APUSH(ICAR)	LISC8890
18	40		ARG = CDR(ARG)	LISC8900
18	41		IF (ARG.LE.0) GO TO 9664	LISC8910
18	42		CALL FPUSH(21,9664)	LISC8920
18	43		GO TO 9662	LISC8930
18	44	9664	CALL APOP(ARG)	LISC8940
18	45		LASTR = CCNS(ARG,LASTR)	LISC8950
18	46		GO TO 998	LISC8960
18	47	9667	IRES= LASTR	LISC8970
18	48		GO TO 998	LISC8980

18	49	9669	IRES=ARG3	LIS08990
18	50		GO TO 998	LIS09000
19	1	C		LIS09010
19	2	C9680	DEFOUT	LIS09020
19	3	9680	IF (ARG.LE.NFREE) GO TO 9933	LIS09030
19	4		LUNO=LUNUT	LIS09040
19	5		LUNUT=ARG-NUMADD	LIS09050
19	6		K=DREG(2)	LIS09060
19	7		DREG(2)=T	LIS09070
19	8		IRES= ARG3	LIS09080
19	9		IF(IRES-NATOM) 9622,9622,9624	LIS09090
19	10	9622	IRES=CONS(IRES,NIL)	LIS09100
19	11	9624	IF (IRES) 9633,9633,9665	LIS09110
19	12	9665	ICAR=CAR(IRES)	LIS09120
19	13		M=CCNS(SUBR+5,CCNS(CONS(CONS(ICAR,CONS(CAR(CDR(CDR(ICAR))))+0,NIL)))	LIS09130
19	14	*	,NIL),CONS(CAR(CDR(ICAR))+0,NIL)))	LIS09140
19	15	9672	CALL IPRINT(M)	LIS09150
19	16	9632	IRES=CDR(IRES)	LIS09160
19	17		GO TO 9624	LIS09170
19	18	9633	LUNUT=LUNO	LIS09180
19	19		DREG(2)=K	LIS09190
19	20		GO TO 998	LIS09200
19	21	C		LIS09210
19	22	C.....		LIS09220
19	23	C0000	SUBR30S AND NOSPREAD SUBR0S	LIS09230
19	24	9300	IF (L-SUBR3) 9309,9309,9400	LIS09240
19	25	C		LIS09250
19	26	C0000	SUBR30S	LIS09260
19	27	9309	ARG=CAR(ARG3)	LIS09270
19	28		IF(ARG3.LE.NATOM.OR.ARG3.GT.NFREE) GO TO 9922	LIS09280
19	29		ICDR = CDR(ARG3)	LIS09290
19	30		IF(ICDR.LE.NATOM.OR.ICDR.GT.NFREE) GO TO 9922	LIS09300
19	31		ARG2 = CAR(ICDR)	LIS09310
19	32		ICDR = CDR(ICDR)	LIS09320
19	33		IF(ICDR.LE.NATOM.OR.ICDR.GT.NFREE) GO TO 9922	LIS09330
19	34		ARG3 = CAR(ICDR)	LIS09340
19	35		L=L-SUBR2	LIS09350
19	36		GOTO (LIS09360
19	37	*	9301, 9302, 9305,	LIS09370
19	38	C	APPLY PUT SASSOC	LIS09380
19	39	*	92011), L	LIS09390
19	40	C		LIS09400
19	41	C9301	APPLY	LIS09410
19	42	9301	I=ARG2	LIS09420
19	43		ARG2=ARG3	LIS09430
19	44		ARG3=I	LIS09440
19	45		GOTO 3000	LIS09450
19	46	C		LIS09460
19	47	C9302	PUT	LIS09470
19	48	9302	K=ARG	LIS09480
19	49	9303	IF (CDR(K)) 9308,9308,9312	LIS09490
19	50	9308	CDR(K)=CCNS(ARG3,CONS(ARG2,NIL))	LIS09500
20	1		GOTO 998	LIS09510
20	2	9312	K=CDR(K)	LIS09520
20	3		IF (CAR(K)-ARG3) 9315,9313,9315	LIS09530
20	4	9313	K=CDR(K)	LIS09540
20	5		IF (K-NFREE) 9314,9314,9311	LIS09550
20	6	9314	CAR(K)=ARG2	LIS09560
20	7		GOTO 998	LIS09570
20	8	9315	K=CDR(K)	LIS09580

20	9		IF (K-NFREE) 9303,9303,9311	LIS09590
20	10	9311	WRITE (3,9319)	LIS09600
20	11	9319	FORMAT (23H PUT OUTSIDE ITS DOMAIN)	LIS09610
20	12		GOTO 998	LIS09620
20	13	C		LIS09630
20	14	C9305	SASSOC	LIS09640
20	15	9305	ARG4=ARG3	LIS09650
20	16		GOTO 7010	LIS09660
20	17	C		LIS09670
20	18	C.....		LIS09680
20	19	C0000	NO-SPREAD SUBROS	LIS09690
20	20	C		LIS09700
20	21	9400	L=L-SUBR3	LIS09710
20	22		GOTO (LIS09720
20	23	*	9402, 9410, 9420, 9430,	LIS09730
20	24	C	LIST PLUS PROG N TIMES	LIS09740
20	25	*	92011), L	LIS09750
20	26	C		LIS09760
20	27	C9402	LIST	LIS09770
20	28	9402	IRES=ARG3	LIS09780
20	29		GOTO 998	LIS09790
20	30	C		LIS09800
20	31	C9410	PLUS	LIS09810
20	32	9410	J=1	LIS09820
20	33		IRES = NUMADD	LIS09830
20	34	9411	IF (ARG3.LE.NATCM.OR.ARG3.GT.NFREE) GO TO 9933	LIS09840
20	35	9413	IF (ARG3) 9933,998,9412	LIS09850
20	36	9412	ICAR=CAR(ARG3)	LIS09860
20	37		IRES = ARITH(J, IRES, ICAR, ARGS AV)	LIS09870
20	38		ARG3=CDR(ARG3)	LIS09880
20	39		GO TO 9413	LIS09890
20	40	C		LIS09900
20	41	C9420	PROGN	LIS09910
20	42	9420	IF (ARG3) 9423,9423,9421	LIS09920
20	43	9421	IF (CDR(ARG3)) 9423,9423,9425	LIS09930
20	44	9423	IRES=CAR(ARG3)	LIS09940
20	45		GOTO 998	LIS09950
20	46	9425	ARG3=CDR(ARG3)	LIS09960
20	47		GOTO 9421	LIS09970
20	48	C		LIS09980
20	49	C9430	TIMES	LIS09990
20	50	9430	IRES= NUMADD+1	LIS10000
21	1		J = 3	LIS10010
21	2		GO TO 9411	LIS10020
21	3	C		LIS10030
21	4	C.....		LIS10040
21	5	C0000	FSUBROS	LIS10050
21	6	C0000	THIS IS A SEPARATE ENTRY POINT (ACCESSED DIRECTLY	LIS10060
21	7	C0000	FROM EVAL). WE ASSUME THAT ARGUMENT LIST AND A-LIST FO	LIS10070
21	8	C0000	THE FSUBR ARE IN ARG VIZ. ARG2, AND THAT THE ATOM INDE	LIS10080
21	9	C0000	IS IN L.	LIS10090
21	10	C		LIS10100
21	11	9591	L=L-SUBR	LIS10110
21	12		GOTO (LIS10120
21	13	*	6000, 9760, 9520, 9550, 9503, 9504, 9505,	LIS10130
21	14	C	COND PROG AND DEFIN DEFLI FUNCTION GO	LIS10140
21	15	*	9530, 9508, 9509,	LIS10150
21	16	C	OR QUOTE SETC	LIS10160
21	17	*	92011), L	LIS10170
21	18	C		LIS10180

21	19	C9550	DEFINE		LIS10190
21	20	9550	ARG3=EXPR		LIS10200
21	21		IF (ARG.LE.NATOM.OR.ARG.GT.NFREE) GO TO 9966		LIS10210
21	22		ARG = CAR(ARG)		LIS10220
21	23	9551	LASTR = 0		LIS10230
21	24		IF (ARG.LE.0) GO TO 9558		LIS10240
21	25		CALL FPUSH(22,9558)		LIS10250
21	26	9552	IF (ARG.LE.NATOM.OR.ARG.GT.NFREE) GO TO 9922		LIS10260
21	27		ICAR= CAR(ARG)		LIS10270
21	28		CALL APUSH(ICAR)		LIS10280
21	29		ARG = CDR(ARG)		LIS10290
21	30		IF (ARG.LE.0) GO TO 9554		LIS10300
21	31		CALL FPUSH(23,9554)		LIS10310
21	32		GO TO 9552		LIS10320
21	33	9554	CALL APOP(ARG)		LIS10330
21	34		IF (ARG.LE.NATOM.OR.ARG.GT.NFREE) GO TO 9922		LIS10340
21	35		ICDR = CDR(ARG)		LIS10350
21	36		IF (ICDR.LE.NATOM.OR.ICDR.GT.NFREE) GO TO 9922		LIS10360
21	37		ARG2 = CAR(ICDR)		LIS10370
21	38		ARG = CAR(ARG)		LIS10380
21	39		CALL FPUSH(24,9556)		LIS10390
21	40		GO TO 9302		LIS10400
21	41	9556	LASTR = CONS(IRES, LASTR)		LIS10410
21	42		GO TO 998		LIS10420
21	43	9558	IRES = LASTR		LIS10430
21	44		GO TO 998		LIS10440
21	45	C			LIS10450
21	46	C9503	DEFLIST		LIS10460
21	47	9503	IF (ARG.LE.NATOM.OR.ARG.GT.NFREE) GO TO 9922		LIS10470
21	48		ICDR = CDR(ARG)		LIS10480
21	49		ARG = CAR(ICDR)		LIS10490
21	50		IF (ICDR.LE.NATOM.OR.ICDR.GT.NFREE) GO TO 9922		LIS10500
22	1		ARG3 = CAR(ICDR)		LIS10510
22	2		GO TO 9551		LIS10520
22	3	C			LIS10530
22	4	C9504	FUNCTION		LIS10540
22	5	9504	ICAR=CAR(ARG)		LIS10550
22	6		IRES=CONS(FUNARG,CONS(ICAR,CONS(ARG2,NIL)))		LIS10560
22	7		GOTO 998		LIS10570
22	8	C			LIS10580
22	9	C9505	GO		LIS10590
22	10	9505	ARG=CAR(ARG)		LIS10600
22	11		GOTO 9740		LIS10610
22	12	C			LIS10620
22	13	C			LIS10630
22	14	C9508	QUOTE		LIS10640
22	15	9508	IRES=CAR(ARG)		LIS10650
22	16		GOTO 998		LIS10660
22	17	C			LIS10670
22	18	C9509	SETQ		LIS10680
22	19	9509	ICAR=CAR(ARG)		LIS10690
22	20		CALL APUSH(ICAR)		LIS10700
22	21		CALL APUSH(ARG2)		LIS10710
22	22		ARG=KAR(CDR(ARG))		LIS10720
22	23		CALL FPUSH(17,9510)		LIS10730
22	24		GOTO 4000		LIS10740
22	25	C@@@	RETURN FRM EVAL		LIS10750
22	26	9510	ARG3=IRES		LIS10760
22	27		CALL APOP(ARG2)		LIS10770
22	28		CALL APOP(ARG)		LIS10780

22	29		GOTC 9218	LIS10790
22	30	C		LIS10800
22	31	C9520	AND	LIS10810
22	32	9520	IF (ARG) 9011,9011,9521	LIS10820
22	33	9521	CALL APUSH(ARG)	LIS10830
22	34		CALL APUSH(ARG2)	LIS10840
22	35	9522	ARG=KAR(STACK(JP+1))	LIS10850
22	36		CALL FPUSH(18,9523)	LIS10860
22	37		GOTO 4000	LIS10870
22	38	C@@@	RETURN FROM EVAL	LIS10880
22	39	9523	IF (IRES) 9528,9528,9524	LIS10890
22	40	9524	K=KDR(STACK(JP+1))	LIS10900
22	41		IF (K) 9528,9528,9525	LIS10910
22	42	9528	JP=JP+2	LIS10920
22	43		GOTO 998	LIS10930
22	44	9525	ARG2=STACK(JP)	LIS10940
22	45		STACK(JP+1)=K	LIS10950
22	46		GOTO 9522	LIS10960
22	47	C		LIS10970
22	48	C9530	OR	LIS10980
22	49	9530	IF (ARG) 9010,9010,9531	LIS10990
22	50	9531	CALL APUSH(ARG)	LIS11000
23	1		CALL APUSH(ARG2)	LIS11010
23	2	9532	ARG=KAR(STACK(JP+1))	LIS11020
23	3		CALL FPUSH(19,9533)	LIS11030
23	4		GOTO 4000	LIS11040
23	5	C@@@	RETURN FROM EVAL	LIS11050
23	6	9533	IF (IRES) 9534,9534,9528	LIS11060
23	7	9534	K=KDR(STACK(JP+1))	LIS11070
23	8		IF (K) 9528,9528,9535	LIS11080
23	9	9535	ARG2=STACK(JP)	LIS11090
23	10		STACK(JP+1)=K	LIS11100
23	11		GOTO 9532	LIS11110
23	12	C.....	LIS11120
23	13	C....	PROG FEATURE	LIS11130
23	14	C		LIS11140
23	15	C9740	GOTO	LIS11150
23	16	9740	IP=IPF	LIS11160
23	17		JP=JPP-5	LIS11170
23	18		ARG2=STACK(JP)	LIS11180
23	19		ARG4=A6	LIS11190
23	20		CALL FPUSH(13,9741)	LIS11200
23	21		GOTO 7000	LIS11210
23	22	C@@@	RETURN FROM CASSOC	LIS11220
23	23	9741	STACK(JP+1)=IRES	LIS11230
23	24		ARG2=STACK(JP+2)	LIS11240
23	25		GOTO 9768	LIS11250
23	26	C		LIS11260
23	27	C9760	PROG	LIS11270
23	28	9760	CALL APUSH(IPP+NUMADD)	LIS11280
23	29		CALL APUSH(JPP+NUMADD)	LIS11290
23	30		IPP=IP	LIS11300
23	31		JPP=JP+2	LIS11310
23	32		ICAR=CAR(ARG)	LIS11320
23	33		ARG2=ADDAL(ICAR,NIL,ARG2,J)	LIS11330
23	34		CALL APUSH(ARG2)	LIS11340
23	35		ICDR=CDR(ARG)	LIS11350
23	36		CALL APUSH(ICDR)	LIS11360
23	37	C0000	PREPARE GO-LIST	LIS11370
23	38		L=CDR(ARG)	LIS11380

23	39		IRES=NIL	LIS11390
23	40	9762	IF (L) 9763,9767,9763	LIS11400
23	41	9763	IF (CAR(L)-NATOM) 9764,9764,9766	LIS11410
23	42	9764	IRES=CONS(L,IRES)	LIS11420
23	43	9766	L=CDR(L)	LIS11430
23	44		GOTO 9762	LIS11440
23	45	9767	CALL APUSH(IRES)	LIS11450
23	46	C0000	PROG LOOP (EVALUATE SUCCESSIVE S-EXPRESSIONS)	LIS11460
23	47	9768	ARG=KAR(STACK(JP+1))	LIS11470
23	48		IF (ARG-NATOM) 9771,9771,9769	LIS11480
23	49	9769	CALL FPUSH(12,9710)	LIS11490
23	50		GOTO 4000	LIS11500
24	1	C0000	RETURN FROM EVAL	LIS11510
24	2	9710	ARG2=STACK(JP+2)	LIS11520
24	3	9771	K=KDR(STACK(JP+1))	LIS11530
24	4		STACK(JP+1)=K	LIS11540
24	5		IF (K) 9768,9780,9768	LIS11550
24	6	C		LIS11560
24	7	C9780	RETURN	LIS11570
24	8	9780	IP=IPF	LIS11580
24	9		JP=JPP-2	LIS11590
24	10		CALL APOP(JPP)	LIS11600
24	11		CALL APOP(IPP)	LIS11610
24	12		JPP=JPP-NUMADD	LIS11620
24	13		IPP=IPP-NUMADD	LIS11630
24	14		IF (IP) 998,9781,998	LIS11640
24	15	9781	WRITE (3,9788)	LIS11650
24	16	9788	FORMAT (20H RETURN OUTSIDE PROG/08H RESTART)	LIS11660
24	17		GOTO 2	LIS11670
24	18	C	-----	LIS11680
24	19	C ERROR MESSAGES	LIS11690
24	20	C		LIS11700
24	21	C9922	INCORRECT NUMBER OF ARGUMENTS	LIS11710
24	22	9922	CALL ERROR(5,ARGSAV,&2)	LIS11720
24	23	C		LIS11730
24	24	C9933	ARGUMENT FORM IMPROPER	LIS11740
24	25	9933	CALL ERROR(6,ARGSAV,&2)	LIS11750
24	26	C		LIS11760
24	27	C9966	THE CAR OF AN ATOM HAS BEEN ATTEMPTED	LIS11770
24	28	9966	CALL ERROR(7,ARGSAV,&2)	LIS11780
24	29		END	LIS11790
24	30		BLOCK DATA	LIS11800
24	31		CCOMMON /CONST/ NATOM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,	LIS11810
24	32	C	LUNUT,IBIG,IRANGE	LIS11820
24	33		COMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC,	LIS11830
24	34	C	IPLUS,IMINUS,IFIG(10),ATEND,ALLSP,CC	LIS11840
24	35		COMMON/ATCM1/PNAME(8192)	LIS11850
24	36		CCOMMON/ATOM2/ PNAME1,PNAME2	LIS11860
24	37		LOGICAL*1 PNAME	LIS11870
24	38		COMMON /INITCM/ ITPAT(8)	LIS11880
24	39		INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT,	LIS11890
24	40	I	ALLSP, RBRACK, UNDEF, ATEND, CC	LIS11900
24	41		DATA ITPAT/10,41,65,68,72,82,92,93/	LIS11910
24	42		LOGICAL*1 PNI(252)/'NIL?ADVANCE?BREAK?EJECT?FORCEGBC?READ?RESTART?LIS11920	
24	43		1SILENCE?TALK?TERPRI?EXIT?ADD1?ATOM?CAR?CDR?CAAR?CADR?CDAR?CDDR?CLELIS11930	
24	44		2ARBIT?GENSYM?INUNIT?NULL?NUMBERP?OUTUNIT?PACKLIST?PRINI?PRINT?PRINLIS11940	
24	45		3TPOS?READPOS?SETBIT?SUB1?TESTBIT?UNPACK?ZEROP?GOTO?RETURN?NOT?MODELIS11950	
24	46		4?OBLIST?'/	LIS11960
24	47		LOGICAL*1 PN2(251)/'CREATE?LOAD?A2?A8?A9?A4?A6?CONS?DIFFEREN?EQ?EQLIS11970	
24	48		1UAL?EVAL?EVLI?GET?GREATERP?LESSP?MEMBER?NCONC?PAIR?QUOTE?IENT?RPLACLIS11980	

```

24 49 2A?RPLACD?SET?MEMB?APPEND?DEFOUT?APPLY?PUT?SASSOC?LIST?PLUS?PROGN?TLIS11990
24 50 3IMES?CCND?PROG?AND?DEFINE?DEFLIST?FUNCTION?GO?OR?QUOTE?SETQ?LAMBDA LIS12000
25 1 4?LABEL?'/ LIS12010
25 2 LOGICAL*1 FN3(42)/'FUNARG?SYS?EXPR?FEXPR?APVAL?SUBR?FSUBR?T??'/ LIS12020
25 3 EQUIVALENCE (PNAME(1),PN1),(PNAME(253),PN2),(PNAME(504),PN3) LIS12030
25 4 C LIS12040
25 5 C NATOM IS THE TOTAL NUMBER OF ATOMS ALLOWED LIS12050
25 6 C NFREET IS THE TOTAL NUMBER OF FREE CELLS IN THE LIST SPACE LIS12060
25 7 C NSTACK IS THE NUMBER OF STACK CELLS LIS12070
25 8 C NATCMP IS THE NUMBER OF SYSTEM ATOMS -1 LIS12080
25 9 C IF THE SYSTEM ATOMS ARE CHANGED *T* MUST BE LAST AND THE LIS12090
25 10 C STATEMENT " T=FSUBR+10 " IN SUBROUTINE INIT MUST BE LIS12100
25 11 C CHANGED TO POINT TO THE ATOM *T* LIS12110
25 12 C IRANGE IS THE LARGEST ABSOLUTE VALUE OF AN INTEGER IT MUST BE LIS12120
25 13 C 2**(N-2) - NFREET/2 -1 LIS12130
25 14 C IBIG IS A LARGE NUMBER THAT FITS IN CAR OR CDR IT MUST BE LIS12140
25 15 C 2**(N-1) - 10 LIS12150
25 16 C NUMADD MUST BE 2**(N-2) + NFREET/2 LIS12160
25 17 C WHERE N IS THE NUMBER OF BITS IN CAR,CDR,STACK LIS12170
25 18 C LIS12180
25 19 DATA NATOM/500/,NFREET/19000/,NSTACK/950/,NATOMP/92/,NUMADD/19034/ LIS12190
25 20 DATA UNDEF/-6000/,LUNIN/1/,LUNUT/3/ LIS12200
25 21 DATA IBIG/32750/,IRANGE/13733/ LIS12210
25 22 DATA SPACE/' '/,LPAR/' ('/,RPAR/')'/',COMMA/',',/,DOT/','/', LIS12220
25 23 C LBRACK/'<'/,RBRACK/'>'/,IPROC/'%'/,IPLUS/'+'/,IMINUS/'-'/, LIS12230
25 24 C IFIG/'0','1','2','3','4','5','6','7','8','9'/,ATEND/'?'/, LIS12240
25 25 C ALLSP/' '/,CC/' '/ LIS12250
25 26 END LIS12260
25 27 FUNCTION MEMB(I,JJ) LIS12270
25 28 COMMON AND INTEGER DECLARATIONS LIS12280
25 29 C----- LIS12290
25 30 COMMON IP, JP, IBR, IPP, JPP, ATENDA, LIS12300
25 31 C NFREEP, NFREEB, LASTR, ARG, ARG2, ARG3, LIS12310
25 32 C ARG4, MODE, ALIST, SUBR0, SUBR1, SUBR2, LIS12320
25 33 C SUBR3, A2, A8, A9, LAMBDA, LABEL, LIS12330
25 34 C FUNARG, NIL, EXPR, FEXPR, SUBR, FSUBR, LIS12340
25 35 C T, SYS, RDPNT, PRTPNT, BSTACK(5), LIS12350
25 36 C DREG(24) LIS12360
25 37 COMMON /CONST/ NATOM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,LUNUT LIS12370
25 38 COMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC, LIS12380
25 39 C IPLUS,IMINUS,IFIG(10),ATEND,ALLSP LIS12390
25 40 COMMON /CARSTR/ NILCAR(2),CAR(10) LIS12400
25 41 COMMON /CDRSTR/ NILCDR(2),CDR(10) LIS12410
25 42 COMMON/ATOM1/ PNAME(4008) LIS12420
25 43 COMMON/ATOM2/ PNAME1,PNAME2 LIS12430
25 44 LOGICAL*1 PNAME LIS12440
25 45 COMMON /STKSTR/ STACK(10) LIS12450
25 46 INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT, LIS12460
25 47 I ALLSP, RBRACK, UNDEF, ATEND, ARG, ARG2, LIS12470
25 48 I ARG3, ARG4, ALIST, SUBR0, SUBR1, SUBR2, LIS12480
25 49 I SUBR3, A2, A8, A9, FUNARG, EXPR, LIS12490
25 50 I FEXPR, SUBR, FSUBR, T, SYS, RDPNT, LIS12500
26 1 I PRTPNT, ATENDA, BSTACK, DREG LIS12510
26 2 INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR LIS12520
26 3 COMMON AND INTEGER DECLARATIONS ENDED LIS12530
26 4 C----- LIS12540
26 5 J=JJ LIS12550
26 6 5 IF (J) 100,100,10 LIS12560
26 7 10 IF (I-CAR(J)) 20,100,20 LIS12570
26 8 20 J=CDR(J) LIS12580

```



```

27 19          COMMON AND INTEGER DECLARATIONS
27 20          C-----
27 21          COMMON  IP,          JP,          IBR,          IPP,          JPP,          ATENDA,
27 22          C      NFREEP,      NFREEB,      LASTR,      ARG,          ARG2,      ARG3,
27 23          C      ARG4,          MODE,      ALIST,      SUBR0,      SUBR1,      SUBR2,
27 24          C      SUBR3,      A2,          A8,          A9,          LAMBDA,    LABEL,
27 25          C      FUNARG,      NIL,        EXPR,      FEXPR,      SUBR,      FSUBR,
27 26          C      T,          SYS,        RDPNT,      PRTPNT,     BSTACK(5),
27 27          C      DREG(24)
27 28          COMMON /CONST/ NATOM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,LUNUTL
27 29          COMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC,
27 30          C IPLUS,IMINUS,IFIG(10),ATEND,ALLSP
27 31          COMMON /CARSTR/ NILCAR(2),CAR(10)
27 32          COMMON /CDRSTR/ NILCDR(2),CDR(10)
27 33          COMMON/ATOM1/ PNAME(4008)
27 34          COMMON/ATCM2/ PNAME1,PNAME2
27 35          LOGICAL*1 PNAME
27 36          COMMON /STKSTR/ STACK(10)
27 37          INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT,
27 38          I      ALLSP, RBRACK, UNDEF, ATEND, ARG, ARG2,
27 39          I      ARG3, ARG4, ALIST, SUBR0, SUBR1, SUBR2,
27 40          I      SUBR3, A2, A8, A9, FUNARG, EXPR,
27 41          I      FEXPR, SUBR, FSUBR, T, SYS, RDPNT,
27 42          I      PRTPNT, ATENDA, BSTACK, DREG
27 43          INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR
27 44          COMMON AND INTEGER DECLARATIONS ENDED
27 45          C-----
27 46          IF (J) 60,60,4
27 47          4 IF (I.EQ.FSUBR+8) GO TO 70
27 48          IF (I.EQ.FSUBR+9) GO TO 80
27 49          K=CDR(J)
27 50          6 IF (K-NFREET) 8,8,40
28 1          8 IF (K) 40,40,10
28 2          10 IF (CAR(K)-I) 12,20,12
28 3          12 K=CDR(K)
28 4          K=CDR(K)
28 5          GOTO 8
28 6          20 K=CDR(K)
28 7          GET=CAR(K)
28 8          RETURN
28 9          40 GET= -1
28 10         RETURN
28 11         60 GET=J
28 12         RETURN
28 13         70 IF (J.LE.SUBR) GO TO 60
28 14         GO TO 40
28 15         80 IF (J.LE.FSUBR.AND.J.GT.SUBR) GO TO 60
28 16         GO TO 40
28 17         END
28 18         SUBROUTINE APUSH(J)
28 19         COMMON AND INTEGER DECLARATIONS
28 20         C-----
28 21         COMMON  IP,          JP,          IBR,          IPP,          JPP,          ATENDA,
28 22         C      NFREEP,      NFREEB,      LASTR,      ARG,          ARG2,      ARG3,
28 23         C      ARG4,          MODE,      ALIST,      SUBR0,      SUBR1,      SUBR2,
28 24         C      SUBR3,      A2,          A8,          A9,          LAMBDA,    LABEL,
28 25         C      FUNARG,      NIL,        EXPR,      FEXPR,      SUBR,      FSUBR,
28 26         C      T,          SYS,        RDPNT,      PRTPNT,     BSTACK(5),
28 27         C      DREG(24)
28 28         COMMON /CONST/ NATOM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,LUNUTL

```

```

28 29      COMMON /SYMBOL/ SPACE, LPAR, RPAR, COMMA, DOT, LBRACK, RBRACK, IPROC,      LIS13790
28 30      C IPLUS, IMINUS, IFIG(10), ATEND, ALLSP      LIS13800
28 31      COMMON /CARSTR/ NILCAR(2), CAR(10)      LIS13810
28 32      COMMON /CDRSTR/ NILCDR(2), CDR(10)      LIS13820
28 33      COMMON /ATOM1/ PNAME(4008)      LIS13830
28 34      COMMON /ATOM2/ PNAME1, PNAME2      LIS13840
28 35      LOGICAL *1 PNAME      LIS13850
28 36      COMMON /STKSTR/ STACK(10)      LIS13860
28 37      INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT,      LIS13870
28 38      I ALLSP, RBRACK, UNDEF, ATEND, ARG, ARG2,      LIS13880
28 39      I ARG3, ARG4, ALIST, SUBR0, SUBR1, SUBR2,      LIS13890
28 40      I SUBR3, A2, A8, A9, FUNARG, EXPR,      LIS13900
28 41      I FEXPR, SUBR, FSUBR, T, SYS, RDPNT,      LIS13910
28 42      I PRTPNT, ATENDA, BSTACK, DREG      LIS13920
28 43      INTEGER *2 CAR, CDR, STACK, NILCAR, NILCDR      LIS13930
28 44      COMMON AND INTEGER DECLARATIONS ENDED      LIS13940
28 45      C-----      LIS13950
28 46      JP=JP-1      LIS13960
28 47      IF (IP-JP) 2,9,9      LIS13970
28 48      2 STACK(JP)=J      LIS13980
28 49      RETURN      LIS13990
28 50      9 WRITE (3,99) JP      LIS14000
29 1      99 FCRMAT (29H ARGUMENTS PDL IS FULL) JP = .I4/ 8H RESTART )      LIS14010
29 2      JP=JP+1      LIS14020
29 3      STACK(IP)=8      LIS14030
29 4      RETURN      LIS14040
29 5      END      LIS14050
29 6      SUBROUTINE FPUSH(I,J)      LIS14060
29 7      COMMON AND INTEGER DECLARATIONS      LIS14070
29 8      C-----      LIS14080
29 9      COMMON IP, JP, IBR, IPP, JPP, ATENDA,      LIS14090
29 10     C NFREEP, NFREEB, LASTR, ARG, ARG2, ARG3,      LIS14100
29 11     C ARG4, MODE, ALIST, SUBR0, SUBR1, SUBR2,      LIS14110
29 12     C SUBR3, A2, A8, A9, LAMBDA, LABEL,      LIS14120
29 13     C FUNARG, NIL, EXPR, FEXPR, SUBR, FSUBR,      LIS14130
29 14     C T, SYS, RDPNT, PRTPNT, BSTACK(5),      LIS14140
29 15     C DREG(24)      LIS14150
29 16     COMMON /CONST/ NATOM, NFREE, NSTACK, NATOMP, NUMADD, UNDEF, LUNIN, LUNUT      LIS14160
29 17     COMMON /SYMBOL/ SPACE, LPAR, RPAR, COMMA, DOT, LBRACK, RBRACK, IPROC,      LIS14170
29 18     C IPLUS, IMINUS, IFIG(10), ATEND, ALLSP      LIS14180
29 19     COMMON /CARSTR/ NILCAR(2), CAR(10)      LIS14190
29 20     COMMON /CDRSTR/ NILCDR(2), CDR(10)      LIS14200
29 21     COMMON /ATOM1/ PNAME(4008)      LIS14210
29 22     COMMON /ATOM2/ PNAME1, PNAME2      LIS14220
29 23     LOGICAL *1 PNAME      LIS14230
29 24     COMMON /STKSTR/ STACK(10)      LIS14240
29 25     INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT,      LIS14250
29 26     I ALLSP, RBRACK, UNDEF, ATEND, ARG, ARG2,      LIS14260
29 27     I ARG3, ARG4, ALIST, SUBR0, SUBR1, SUBR2,      LIS14270
29 28     I SUBR3, A2, A8, A9, FUNARG, EXPR,      LIS14280
29 29     I FEXPR, SUBR, FSUBR, T, SYS, RDPNT,      LIS14290
29 30     I PRTPNT, ATENDA, BSTACK, DREG      LIS14300
29 31     INTEGER *2 CAR, CDR, STACK, NILCAR, NILCDR      LIS14310
29 32     COMMON AND INTEGER DECLARATIONS ENDED      LIS14320
29 33     C-----      LIS14330
29 34     IP=IP+1      LIS14340
29 35     IF (IP-JP) 2,9,9      LIS14350
29 36     2 STACK(IF)=I      LIS14360
29 37     RETURN      LIS14370
29 38     9 WRITE (3,99) IP      LIS14380

```

```

29 39          99 FORMAT (28H FUNCTION PDL IS FULL| IP = , I4/ 8H RESTART ) LIS14390
29 40          IP=IP-1 LIS14400
29 41          STACK(IF)=8 LIS14410
29 42          RETURN LIS14420
29 43          END LIS14430
29 44          SUBROUTINE APOP(J) LIS14440
29 45          CCMCN AND INTEGER DECLARATIONS LIS14450
29 46          C----- LIS14460
29 47          CCMCN IP, JP, IBR, IPP, JPP, ATENDA, LIS14470
29 48          C NFREEP, NFREEB, LASTR, ARG, ARG2, ARG3, LIS14480
29 49          C ARG4, MODE, ALIST, SUBR0, SUBR1, SUBR2, LIS14490
29 50          C SUBR3, A2, A8, A9, LAMBDA, LABEL, LIS14500
30 1          C FUNARG, NIL, EXPR, FEXPR, SUBR, FSUBR, LIS14510
30 2          C T, SYS, RDPNT, PRTPNT, BSTACK(5), LIS14520
30 3          C DREG(24) LIS14530
30 4          COMMON /CONST/ NATOM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,LUNUT LIS14540
30 5          COMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC, LIS14550
30 6          C IPLUS,IMINUS,IFIG(10),ATEND,ALLSP LIS14560
30 7          CCMCN /CARSTR/ NILCAR(2),CAR(10) LIS14570
30 8          COMMON /CDRSTR/ NILCDR(2),CDR(10) LIS14580
30 9          CCMCN/ATCM1/ PNAME(4008) LIS14590
30 10         CCMCN/ATOM2/ PNAME1,PNAME2 LIS14600
30 11         LOGICAL*1 PNAME LIS14610
30 12         CCMCN /STKSTR/ STACK(10) LIS14620
30 13         INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT, LIS14630
30 14         I ALLSP, RERACK, UNDEF, ATEND, ARG, ARG2, LIS14640
30 15         I ARG3, ARG4, ALIST, SUBR0, SUBR1, SUBR2, LIS14650
30 16         I SUBR3, A2, A8, A9, FUNARG, EXPR, LIS14660
30 17         I FEXPR, SUBR, FSUBR, T, SYS, RDPNT, LIS14670
30 18         I PRTPNT, ATENDA, BSTACK, DREG LIS14680
30 19         INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR LIS14690
30 20         CCMCN AND INTEGER DECLARATIONS ENDED LIS14700
30 21         C----- LIS14710
30 22         JP=JP+1 LIS14720
30 23         IF (JP-NSTACK) 2,2,9 LIS14730
30 24         2 J=STACK(JP-1) LIS14740
30 25         RETURN LIS14750
30 26         9 WRITE (3,99) LIS14760
30 27         99 FCFMAT (19H ARG PDL IS EMPTY.?) LIS14770
30 28         STACK(IF)=8 LIS14780
30 29         RETURN LIS14790
30 30         END LIS14800
30 31         SUBROUTINE PRINAT(I,IATOM) LIS14810
30 32         CCMCN AND INTEGER DECLARATIONS LIS14820
30 33         C----- LIS14830
30 34         COMMON IP, JP, IBR, IPP, JPP, ATENDA, LIS14840
30 35         C NFREEP, NFREEB, LASTR, ARG, ARG2, ARG3, LIS14850
30 36         C ARG4, MODE, ALIST, SUBR0, SUBR1, SUBR2, LIS14860
30 37         C SUBR3, A2, A8, A9, LAMBDA, LABEL, LIS14870
30 38         C FUNARG, NIL, EXPR, FEXPR, SUBR, FSUBR, LIS14880
30 39         C T, SYS, RDPNT, PRTPNT, BSTACK(5), LIS14890
30 40         C DREG(24) LIS14900
30 41         COMMON /CONST/ NATCM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,LUNUT LIS14910
30 42         COMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC, LIS14920
30 43         C IPLUS,IMINUS,IFIG(10),ATEND,ALLSP LIS14930
30 44         CCMCN /CARSTR/ NILCAR(2),CAR(10) LIS14940
30 45         CCMCN /CDRSTR/ NILCDR(2),CDR(10) LIS14950
30 46         CCMCN/ATOM1/ PNAME(4008) LIS14960
30 47         CCMCN/ATOM2/ FNAME1,PNAME2 LIS14970
30 48         LOGICAL*1 PNAME LIS14980

```

30	49	COMMON /STKSTR/ STACK(10)							LIS14990
30	50	INTEGER PNAME1, PNAME2,	SPACE,	RPAR,	COMMA,	DOT,		LIS15000	
31	1	I ALLSP, RBRACK,	UNDEF,	ATEND,	ARG,	ARG2,		LIS15010	
31	2	I ARG3, ARG4,	ALIST,	SUBR0,	SUBR1,	SUBR2,		LIS15020	
31	3	I SUBR3, A2,	A8,	A9,	FUNARG,	EXPR,		LIS15030	
31	4	I FEXPR, SUBR,	FSUBR,	T,	SYS,	RDPNT,		LIS15040	
31	5	I PRTPNT, ATENDA,	BSTACK,	DREG				LIS15050	
31	6	INTEGER PNAMEP,PNAMEN						LIS15060	
31	7	INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR						LIS15070	
31	8	CCMMCN AND INTEGER DECLARATIONS ENDED						LIS15080	
31	9	C-----						LIS15090	
31	10	C PRINAT PRINTS ONE ATCM OR ONE CHARACTER DEPENDING ON I						LIS15100	
31	11	C I=1 PRINT (LIS15110	
31	12	C I=2 PRINT)						LIS15120	
31	13	C I=3 PRINT .ATOM)						LIS15130	
31	14	C I=4 PRINT ATOM						LIS15140	
31	15	C IND = CHARACTERPOINTER TO OUTPUTBUFFER Ibuff						LIS15150	
31	16	DIMENSION Ibuff(18),IPDUM(1),PRST(10),ISIFF(8)						LIS15160	
31	17	DATA Ibuff/18*' /						LIS15170	
31	18	INTEGER RPARIN,PRST						LIS15180	
31	19	EQUIVALENCE (IND,PRTPNT)						LIS15190	
31	20	DATA IFIRST/0/						LIS15200	
31	21	IF(1) 804,804,30						LIS15210	
31	22	30 IF(IFIRST-9) 40,50,40						LIS15220	
31	23	40 NPRST=0						LIS15230	
31	24	IPARIN=0						LIS15240	
31	25	IFIRST=9						LIS15250	
31	26	ILAST=0						LIS15260	
31	27	IBUFF(1)=ALLSP						LIS15270	
31	28	IBUFF(2)=ALLSP						LIS15280	
31	29	DO 45 J=1,10						LIS15290	
31	30	45 PRST(J)=0						LIS15300	
31	31	50 GO TO (100,200,300,400),I						LIS15310	
31	32	C () .A) A						LIS15320	
31	33	C0000 (00000000000000000000000000000000						LIS15330	
31	34	100 IF(ILAST-2) 120,105,120						LIS15340	
31	35	105 IF(DREG(2)) 120,110,120						LIS15350	
31	36	110 CALL OUTSTR(IBUFF,IPARIN,IND)						LIS15360	
31	37	120 IF(NPRST) 140,140,130						LIS15370	
31	38	130 PRST(NPRST)=PRST(NPRST)+1						LIS15380	
31	39	140 CALL PCHAR(IBUFF,LPAR,IND)						LIS15390	
31	40	IPARIN=IPARIN+1						LIS15400	
31	41	GO TO 800						LIS15410	
31	42	C0000) 00000000000000000000000000000000						LIS15420	
31	43	200 IF(NPRST) 240,240,210						LIS15430	
31	44	210 PRST(NPRST)=PRST(NPRST)-1						LIS15440	
31	45	IF(PRST(NPRST)) 230,230,240						LIS15450	
31	46	230 NPRST=NPRST-1						LIS15460	
31	47	240 CALL PCHAR(IBUFF,RPAR,IND)						LIS15470	
31	48	IPARIN=IPARIN-1						LIS15480	
31	49	GO TO 800						LIS15490	
31	50	C0000 .ATOM) 000000000000000000000000000000						LIS15500	
32	1	300 CALL PCHAR(IBUFF,DOT,IND)						LIS15510	
32	2	320 IND=IND+1						LIS15520	
32	3	CALL PCHAR(IBUFF,SPACE,IND)						LIS15530	
32	4	325 RPARIN=1						LIS15540	
32	5	IPARIN=IPARIN-1						LIS15550	
32	6	IF(NPRST) 420,420,3250						LIS15560	
32	7	3250 PRST(NPRST)=PRST(NPRST)-1						LIS15570	
32	8	IF(PRST(NPRST)) 3275,3275,420						LIS15580	

32	9	3275	NPRST=NFRST-1	LIS15590
32	10		GO TO 420	LIS15600
32	11	C0000	ATCM 000000000000000000000000000000	LIS15610
32	12	400	IF(IND-60) 420,410,410	LIS15620
32	13	410	CALL OUTSTR(IBUFF,IPARIN,IND)	LIS15630
32	14	C0000	TEST IF LAMBDA OR PROG	LIS15640
32	15	420	IF(DREG(2)) 460,425,460	LIS15650
32	16	425	IF(LAMBDA-IATCM) 440,430,440	LIS15660
32	17	430	PRST(NPRST)=PRST(NPRST)-1	LIS15670
32	18		NPRST=NPRST+1	LIS15680
32	19		PRST(NPRST)=1	LIS15690
32	20		GO TO 460	LIS15700
32	21	440	IF(SUBR+2-IATOM) 450,430,450	LIS15710
32	22	C..	PROG SHULD BE SUBR+2.	LIS15720
32	23	C0000	NOW TEST IF ATCM-LABEL IN PROG.	LIS15730
32	24	450	IF(PRST(NPRST)-1) 460,455,460	LIS15740
32	25	455	CALL OUTSTR(IBUFF,IPARIN,IND)	LIS15750
32	26	C0000	NOW TEST IF NUM. ATOM	LIS15760
32	27	460	IF(NATCM-IATCM) 465,470,470	LIS15770
32	28	C0000	NUM. ATCM0000000000000000000000000000	LIS15780
32	29	465	IATO=IATOM-NUMADD	LIS15790
32	30		IF(IATO) 466,467,467	LIS15800
32	31	466	CALL PCHAR(IBUFF,IMINUS,IND)	LIS15810
32	32		IATO=-IATC	LIS15820
32	33		IND=IND+1	LIS15830
32	34	467	ISI=1	LIS15840
32	35		DC 468 J=1,8	LIS15850
32	36		JJ=IATC-IATC/10*10	LIS15860
32	37		ISIFF(J)=IFIG(JJ+1)	LIS15870
32	38		IATO=IATO/10	LIS15880
32	39		IF(IATC) 4675,468,4675	LIS15890
32	40	4675	ISI=ISI+1	LIS15900
32	41	468	CONTINUE	LIS15910
32	42		DD 469 J=1,ISI	LIS15920
32	43		J9=ISI+1-J	LIS15930
32	44		CALL PCHAR(IBUFF,ISIFF(J9),IND)	LIS15940
32	45		IND=IND+1	LIS15950
32	46	469	CONTINUE	LIS15960
32	47		GO TO 4200	LIS15970
32	48	C0000	NOT NUM. ATCM 0000000000000000000000	LIS15980
32	49	470	CONTINUE	LIS15990
32	50		LOGICAL*1 LIBUFF(1)	LIS16000
33	1		EQUIVALENCE (IBUFF,LIBUFF)	LIS16010
33	2		CALL MVC(LIBUFF(IND),PNAME(PNAMEP(IATOM)),PNAME2)	LIS16020
33	3		IND=IND+PNAME2	LIS16030
33	4	C0000	ATCM READY. TEST IF RPARIN OR PROG-LABEL	LIS16040
33	5	4200	CALL PCHAR(IBUFF,SPACE,IND)	LIS16050
33	6		IND=IND+1	LIS16060
33	7		IF(RPARIN-1) 4300,4210,4300	LIS16070
33	8	4210	CALL PCHAR(IBUFF,RPAR,IND)	LIS16080
33	9		RPARIN=0	LIS16090
33	10		GO TO 800	LIS16100
33	11	C00000	TEST IF LAMBDA OR PROG. IN THAT CASE NO NEW LINE.	LIS16110
33	12	4300	IF(DREG(2)) 801,4310,801	LIS16120
33	13	4310	IF(SUBR+2-IATOM) 4320,801,4320	LIS16130
33	14	4320	IF(LAMBDA-IATCM) 4325,801,4325	LIS16140
33	15	C00000	TEST IF NEW LINE AFTER ATOM-LABEL IN PROG	LIS16150
33	16	4325	IF(PRST(NPRST)-1) 801,4350,801	LIS16160
33	17	4350	CALL OUTSTR(IBUFF,IPARIN,IND)	LIS16170
33	18		GO TO 801	LIS16180

```

33 19      800 IND=IND+1                                LIS16190
33 20      801 ILAST=I                                LIS16200
33 21      IF(IND-60) E10,810,805                    LIS16210
33 22      804 IFIRST=0                                LIS16220
33 23      805 CALL OUTSTR(IBUFF,IPAR IN,IND)          LIS16230
33 24      810 RETURN                                  LIS16240
33 25      END                                          LIS16250
33 26      SUBROUTINE CUTSTR(IBUFF,IBL,IND)            LIS16260
33 27      COMMON AND INTEGER DECLARATIONS            LIS16270
33 28      C-----*-----*-----*-----*-----* LIS16280
33 29      COMMON IP, JP, IBR, IPP, JPP, ATENDA, LIS16290
33 30      C NFREEP, NFREEB, LASTR, ARG, ARG2, ARG3, LIS16300
33 31      C ARG4, MODE, ALIST, SUBR0, SUBR1, SUBR2, LIS16310
33 32      C SUBR3, A2, A8, A9, LAMBDA, LABEL, LIS16320
33 33      C FUNARG, NIL, EXPR, FEXPR, SUBR, FSUBR, LIS16330
33 34      C T, SYS, RDPNT, PRTPNT, BSTACK(5), LIS16340
33 35      C DREG(24) LIS16350
33 36      COMMON /CONST/ NATOM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,LUNUT LIS16360
33 37      COMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC, LIS16370
33 38      C IPLUS,IMINUS,IFIG(10),ATEND,ALLSP LIS16380
33 39      COMMON /CARSTR/ NILCAR(2),CAR(10) LIS16390
33 40      COMMON /CDRSTR/ NILCDR(2),CDR(10) LIS16400
33 41      COMMON/ATOM1/ PNAME(4008) LIS16410
33 42      COMMON/ATOM2/ PNAME1,PNAME2 LIS16420
33 43      LOGICAL*1 PNAME LIS16430
33 44      COMMON /STKSTR/ STACK(10) LIS16440
33 45      INTEGER PNAME1, FNAME2, SPACE, RPAR, COMMA, DOT, LIS16450
33 46      I ALLSP, RBRACK, UNDEF, ATEND, ARG, ARG2, LIS16460
33 47      I ARG3, ARG4, ALIST, SUBR0, SUBR1, SUBR2, LIS16470
33 48      I SUBR3, A2, A8, A9, FUNARG, EXPR, LIS16480
33 49      I FEXPR, SUBR, FSUBR, T, SYS, RDPNT, LIS16490
33 50      I PRTPNT, ATENDA, BSTACK, DREG LIS16500
34 1      INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR LIS16510
34 2      COMMON AND INTEGER DECLARATIONS ENDED LIS16520
34 3      C-----*-----*-----*-----* LIS16530
34 4      DIMENSION IBUFF(18) LIS16540
34 5      DATA MAX/18/ LIS16550
34 6      C TO MAKE OUTPUT LINE OF MIN. LENGTH REMOVE ABOVE CARD AND LIS16560
34 7      C CHANGE FOLLOWING CARD TO EXECUTABLE LIS16570
34 8      C LIS16580
34 9      MAX=IND/4+1 LIS16590
34 10     WRITE(LUNUT,100) (IBUFF(I),I=1,MAX) LIS16600
34 11     DO 1 I=1,18 LIS16610
34 12     1 IBUFF(I)=ALLSP LIS16620
34 13     IF(DREG(2)) 3,2,3 LIS16630
34 14     2 IND=2*IEL+1 LIS16640
34 15     RETURN LIS16650
34 16     3 IND=1 LIS16660
34 17     RETURN LIS16670
34 18     100 FORMAT(1X,18A4) LIS16680
34 19     END LIS16690
34 20     SUBROUTINE PCHAR(IEBUFF,NP1,IND) LIS16700
34 21     DIMENSION IEBUFF(18) LIS16710
34 22     I=(IND-1)/4+1 LIS16720
34 23     J=IND-(IND-1)/4*4 LIS16730
34 24     CALL PUTCH(IEBUFF(I),NP1,J) LIS16740
34 25     RETURN LIS16750
34 26     END LIS16760
34 27     SUBROUTINE IPRINT(IJK) LIS16770
34 28     COMMON AND INTEGER DECLARATIONS LIS16780
34 29     C-----*-----*-----*-----*

```

```

34 29          COMMON  IP,          JP,          IBR,          IPP,          JPP,          ATENDA, LIS16790
34 30          C          NFREEP,    NFREEB,    LASTR,    ARG,    ARG2,    ARG3,    LIS16800
34 31          C          ARG4,    MODE,    ALIST,    SUBR0,    SUBR1,    SUBR2,    LIS16810
34 32          C          SUBR3,    A2,    A8,    A9,    LAMBDA,    LABEL,    LIS16820
34 33          C          FUNARG,    NIL,    EXPR,    FEXPR,    SUBR,    FSUBR,    LIS16830
34 34          C          T,    SYS,    RDPNT,    PRTPNT,    BSTACK(5),    LIS16840
34 35          C          DREG(24)    LIS16850
34 36          COMMON /CONST/ NATOM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,    LIS16860
34 37          C LUNUT,IBIG,IRANGE    LIS16870
34 38          COMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC,    LIS16880
34 39          C IPLUS,IMINUS,IFIG(10),ATEND,ALLSP    LIS16890
34 40          COMMON /CARSTR/ NILCAR(2),CAR(10)    LIS16900
34 41          COMMON /CDRSTR/ NILCDR(2),CDR(10)    LIS16910
34 42          COMMON/ATOM1/ FNAME(40C8)    LIS16920
34 43          COMMON/ATOM2/ PNAME1,PNAME2    LIS16930
34 44          LOGICAL*1 PNAME    LIS16940
34 45          COMMON /STKSTR/ STACK(10)    LIS16950
34 46          INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT,    LIS16960
34 47          I ALLSP, RBRACK, UNDEF, ATEND, ARG, ARG2,    LIS16970
34 48          I ARG3, ARG4, ALIST, SUBR0, SUBR1, SUBR2,    LIS16980
34 49          I SUBR3, A2, A8, A9, FUNARG, EXPR,    LIS16990
34 50          I FEXPR, SUBR, FSUBR, T, SYS, RDPNT,    LIS17000
35 1          I PRTPNT, ATENDA, BSTACK, DREG    LIS17010
35 2          INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR    LIS17020
35 3          COMMON AND INTEGER DECLARATIONS ENDED    LIS17030
35 4          C -----    LIS17040
35 5          C    LIS17050
35 6          C          NO-RECURSIVE PRINT    LIS17060
35 7          C          IUP POINTS TO THE UPPER CELL    LIS17070
35 8          C          IDCWN POINTS TO THE LOWER CELL    LIS17080
35 9          C          I POINTS TO THE CELL    LIS17090
35 10         C          USES THE IDEA OF GARBAGE-COLL., BUT    LIS17100
35 11         C          SEARCHES ALONG CAR INSTEAD OF CDR    LIS17110
35 12         C    LIS17120
35 13         C          FIRST TEST IF LIST OR NOT    LIS17130
35 14         C          IDUM=0    LIS17140
35 15         C          I=IJK    LIS17150
35 16         C          IF(I) 6,3,1    LIS17160
35 17         C          1 IF(I-NATOM) 3,3,2    LIS17170
35 18         C          2 IF(NFREET-I)3,4,4    LIS17180
35 19         C          3 CALL PRINAT(4,I)    LIS17190
35 20         C          CALL PRINAT(-1,IDUM)    LIS17200
35 21         C          6 RETURN    LIS17210
35 22         C          4 IVAL=I    LIS17220
35 23         C          IUP= IBIG    LIS17230
35 24         C          CALL PRINAT(1,IDUM)    LIS17240
35 25         C          C0000 DOWN SCAN 000000000000000000000000000000000000    LIS17250
35 26         C          5 IF(CDR(I)) 20,9,9    LIS17260
35 27         C          9 IF(CAR(I)) 20,10,10    LIS17270
35 28         C          10 CDR(I)=-CDR(I)-1    LIS17280
35 29         C          IDCWN=CAR(I)    LIS17290
35 30         C          CAR(I)=IUP    LIS17300
35 31         C          IF(IDOWN-NATOM) 25,15,15    LIS17310
35 32         C          15 IF(IDOWN-NFREET) 16,16,25    LIS17320
35 33         C          16 CALL PRINAT(1,IDUM)    LIS17330
35 34         C          C0000 PRINT ( 0000000000    LIS17340
35 35         C          17 IUP=I    LIS17350
35 36         C          I=IDOWN    LIS17360
35 37         C          GO TO 5    LIS17370
35 38         C          20 I=IUP    LIS17380

```

```

35 39      GO TO 50
35 40      25 CALL PRINAT(4, IDOWN)
35 41      C000 PRINT ATOM C000000000
35 42      GO TO 50
35 43      C
35 44      C0000 REVERSE SCAN 00000000000000000000000000000000
35 45      C
35 46      50 IF(I- IBIG ) 55,95,55
35 47      55 IF(CDR(I)- IBIG ) 60,90,60
35 48      60 IF(CAR(I)) 75,70,70
35 49      C 000 NC MARK FOR BRANCH-POINT000
35 50      70 IUP=CAR(I)
36 1      CAR(I)=IDOWN
36 2      GO TO 80
36 3      C0000 MARK FOR BRANCH-POINT FOUND
36 4      75 ISL=I
36 5      I=CDR(I)
36 6      CDR(ISL)=IDCWN
36 7      CAR(ISL)=-CAR(ISL)-1
36 8      IDCWN=ISL
36 9      GO TO 50
36 10     C0000 TEST IF CDR(I)= LIST
36 11     C0000 OBS. CDR(I) IS NEGATIVE HERE
36 12      80 IF(NATCM+CDR(I)+1)81,85,85
36 13      81 IF(-1-CCR(I)-NFREET) 82,82,85
36 14     C0000 CDR(I) DOES POINT TO A LIST 000
36 15      82 IDOWN=-CDR(I)-1
36 16      CDR(I)=IUP
36 17      CAR(I)=-CAR(I)-1
36 18      GO TO 17
36 19     C0000 CDR(2) DOES NOT POINT TO A LIST
36 20      85 CDR(I)=-CDR(I)-1
36 21      IF(CDR(I)) 87,86,87
36 22      86 CALL PRINAT(2, IDUM)
36 23     C0000 PRINT ) 000000
36 24      GO TO 88
36 25      87 ICDR=CDR(I)
36 26      CALL PRINAT(3, ICDR)
36 27     C0000 PRINT ,ATOM)
36 28      88 IDOWN=I
36 29      I=IUP
36 30      GO TO 50
36 31      90 CDR(I)=IDOWN
36 32      CAR(I)=-CAR(I)-1
36 33      95 I=IVAL
36 34      CALL PRINAT(-1, IDUM)
36 35      PRTPNT=1
36 36      RETURN
36 37      END
36 38      SUBROUTINE INIT(IRESTA)
36 39      CCOMMON AND INTEGER DECLARATIONS
36 40      C-----
36 41      CCOMMON IP, JP, IBR, IPP, JPP, ATENDA,
36 42      C NFREEP, NFREEB, LASTR, ARG, ARG2, ARG3,
36 43      C ARG4, MODE, ALIST, SUBR0, SUBR1, SUBR2,
36 44      C SUBR3, A2, A8, A9, LAMBDA, LABEL,
36 45      C FUNARG, NIL, EXPR, FEXPR, SUBR, FSUBR,
36 46      C T, SYS, RDPNT, PRTPNT, BSTACK(5),
36 47      C DREG(24), APVAL
36 48      COMMON /CONST/ NATOM, NFREET, NSTACK, NATOMP, NUMADD, UNDEF, LUNIN, LUNUTL

```

```

LIS17390
LIS17400
LIS17410
LIS17420
LIS17430
LIS17440
LIS17450
LIS17460
LIS17470
LIS17480
LIS17490
LIS17500
LIS17510
LIS17520
LIS17530
LIS17540
LIS17550
LIS17560
LIS17570
LIS17580
LIS17590
LIS17600
LIS17610
LIS17620
LIS17630
LIS17640
LIS17650
LIS17660
LIS17670
LIS17680
LIS17690
LIS17700
LIS17710
LIS17720
LIS17730
LIS17740
LIS17750
LIS17760
LIS17770
LIS17780
LIS17790
LIS17800
LIS17810
LIS17820
LIS17830
LIS17840
LIS17850
LIS17860
LIS17870
LIS17880
LIS17890
LIS17900
LIS17910
LIS17920
LIS17930
LIS17940
LIS17950
LIS17960
LIS17970
LIS17980

```



```

36 49      COMMON /SYMBOL/ SPACE, LPAR, RPAR, COMMA, DOT, LBRACK, RBRACK, IPROC, LIS17990
36 50      C IPLUS, IMINUS, IFIG(10), ATEND, ALLSP LIS18000
37 1        COMMON /CARSTR/ NILCAR(2), CAR(10) LIS18010
37 2        COMMON /CDRSTR/ NILCDR(2), CDR(10) LIS18020
37 3        COMMON /ATOM1/ FNAME(4008) LIS18030
37 4        COMMON /ATOM2/ PNAME1, PNAME2 LIS18040
37 5        LCGICAL*1 PNAME, NE LIS18050
37 6        COMMON /STKSTR/ STACK(10) LIS18060
37 7        INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT, LIS18070
37 8        I ALLSP, RBRACK, UNDEF, ATEND, ARG, ARG2, LIS18080
37 9        I ARG3, ARG4, ALIST, SUBR0, SUBR1, SUBR2, LIS18090
37 10       I SUBR3, A2, A8, A9, FUNARG, EXPR, LIS18100
37 11       I FEXPR, SUBR, FSUBR, T, SYS, RDPNT, LIS18110
37 12       I PRTPNT, ATENDA, BSTACK, DREG, APVAL LIS18120
37 13       INTEGER PNAMEP, PNAMEN LIS18130
37 14       INTEGER*2 CAR, CDR, STACK, NILCAR, NILCDR LIS18140
37 15       COMMON AND INTEGER DECLARATIONS ENDED LIS18150
37 16       C----- LIS18160
37 17       COMMON /INITCM/ ITYPAT(8) LIS18170
37 18       INTEGER CCNS, FATOM LIS18180
37 19       C LIS18190
37 20       C** INIT GIVES INITIAL-VALUES TO SYSTEM CONSTANTS LIS18200
37 21       MAX = NATCMP LIS18210
37 22       NFREEB = NATCM + 1 LIS18220
37 23       NFREEP = NFREEB LIS18230
37 24       IP = 0 LIS18240
37 25       JP = NSTACK + 1 LIS18250
37 26       IBR = 0 LIS18260
37 27       IPP = IP LIS18270
37 28       JFP = JP LIS18280
37 29       PRTPNT = 1 LIS18290
37 30       IF (IRESTA) 21, 2, 2 LIS18300
37 31       2 CONTINUE LIS18310
37 32       ATENDA = SPACE LIS18320
37 33       DO 25 I = 1, 4 LIS18330
37 34       25 CALL PUTCH(ATENDA, ATEND, I) LIS18340
37 35       21 SUBR0 = ITYPAT(1) LIS18350
37 36       SUBR1 = ITYPAT(2) LIS18360
37 37       SUBR2 = ITYPAT(3) LIS18370
37 38       SUBR3 = ITYPAT(4) LIS18380
37 39       SUBR = ITYPAT(5) LIS18390
37 40       FSUBR = ITYPAT(6) LIS18400
37 41       A2 = SUBR1 + 1 LIS18410
37 42       A8 = SUBR1 + 2 LIS18420
37 43       A9 = SUBR1 + 3 LIS18430
37 44       LAMBDA = FSUBR + 1 LIS18440
37 45       LABEL = FSUBR + 2 LIS18450
37 46       FUNARG = FSUBR + 3 LIS18460
37 47       SYS = FSUBR + 4 LIS18470
37 48       EXPR = FSUBR + 5 LIS18480
37 49       FEXPR = FSUBR + 6 LIS18490
37 50       APVAL = FSUBR + 7 LIS18500
38 1        T = FSUBR + 10 LIS18510
38 2        ARG = 0 LIS18520
38 3        ARG2 = 0 LIS18530
38 4        ARG3 = 0 LIS18540
38 5        ARG4 = 0 LIS18550
38 6        DO 27 J = 1, 2 LIS18560
38 7        CAR(J-2) = 0 LIS18570
38 8        27 CDR(J-2) = 0 LIS18580

```

```

38 9          NILCAR(2)=-1
38 10         J=0
38 11         MAX=MAX+1
38 12         28 DO 30 I=1,MAX
38 13         CDR(I)=0
38 14         29 J=J+1
38 15         IF(NE(PNAME(J),'?'))GO TO 29
38 16         30 CAR(I)=-{J+1}
38 17         MAX=MAX+1
38 18         DO 32 I=MAX,NFREET
38 19         32 CAR(I)=0
38 20         LASTR=0
38 21         NIL=0
38 22         CALL MAKFRE
38 23         IDUM=RATOM(-1)
38 24         CALL SHIFT(123456)
38 25         CALL JATCM(SPACE,-3)
38 26         CDR(T)=CONS(APVAL,CONS(T,NIL))
38 27         ALIST=CCONS(CONS(NIL,NIL),NIL)
38 28         MODE=2
38 29         RETURN
38 30         END
38 31         SUBROUTINE MAKFRE
38 32         COMMON AND INTEGER DECLARATIONS
38 33         C-----
38 34         CCOMMON IP, JP, IBR, IPP, JPP, ATENDA,
38 35         C NFREEP, NFREEB, LASTR, ARG, ARG2, ARG3,
38 36         C ARG4, MODE, ALIST, SUBR0, SUBR1, SUBR2,
38 37         C SUBR3, A2, A8, A9, LAMBDA, LABEL,
38 38         C FUNARG, NIL, EXPR, FEXPR, SUBR, FSUBR,
38 39         C T, SYS, RDPNT, PRTPNT, BSTACK(5),
38 40         C DREG(24)
38 41         COMMON /CONST/ NATOM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,LUNUT
38 42         COMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC,
38 43         C IPLUS,IMINUS,IFIG(10),ATEND,ALLSP
38 44         CCOMMON /CARSTR/ NILCAR(2),CAR(10)
38 45         CCOMMON /CDRSTR/ NILCDR(2),CDR(10)
38 46         COMMON/ATOM1/ PNAME(4008)
38 47         CCOMMON/ATCM2/ PNAME1,PNAME2
38 48         LOGICAL*1 FNAME
38 49         CCOMMON /STKSTR/ STACK(10)
38 50         INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT,
39 1         I ALLSP, RBRACK, UNDEF, ATEND, ARG, ARG2,
39 2         I ARG3, ARG4, ALIST, SUBR0, SUBR1, SUBR2,
39 3         I SUBR3, A2, A8, A9, FUNARG, EXPR,
39 4         I FEXPR, SUBR, FSUBR, T, SYS, RDPNT,
39 5         I PRTPNT, ATENCA, BSTACK, DREG
39 6         INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR
39 7         COMMON AND INTEGER DECLARATIONS ENDED
39 8         C-----
39 9         C0000 MAKFRE GENERATES A FREE LIST
39 10        C0000 THE VALUE IS IN NFREEP
39 11        NFREEP=0
39 12        DO 100 I=NFREEB,NFREET
39 13        IF(CAR(I))10,15,15
39 14        10 CAR(I)=-CAR(I)-1
39 15        GO TO 100
39 16        15 IF(NFREEP) 17,16,17
39 17        16 NFREEP=I
39 18        ILAST=I

```

```

LIS18590
LIS18600
LIS18610
LIS18620
LIS18630
LIS18640
LIS18650
LIS18660
LIS18670
LIS18680
LIS18690
LIS18700
LIS18710
LIS18720
LIS18730
LIS18740
LIS18750
LIS18760
LIS18770
LIS18780
LIS18790
LIS18800
LIS18810
LIS18820
LIS18830
LIS18840
LIS18850
LIS18860
LIS18870
LIS18880
LIS18890
LIS18900
LIS18910
LIS18920
LIS18930
LIS18940
LIS18950
LIS18960
LIS18970
LIS18980
LIS18990
LIS19000
LIS19010
LIS19020
LIS19030
LIS19040
LIS19050
LIS19060
LIS19070
LIS19080
LIS19090
LIS19100
LIS19110
LIS19120
LIS19130
LIS19140
LIS19150
LIS19160
LIS19170
LIS19180

```

```

39 19          CDR(I)=NIL                               LIS19190
39 20          GO TO 100                               LIS19200
39 21          17 CDR(ILAST)=I                          LIS19210
39 22          ILAST=I                                LIS19220
39 23          100 CONTINUE                             LIS19230
39 24          CDR(ILAST)=NIL                          LIS19240
39 25          RETURN                                  LIS19250
39 26          END                                     LIS19260
39 27          INTEGER FUNCTION CCNS(I1,I2)            LIS19270
39 28          COMMON AND INTEGER DECLARATIONS        LIS19280
39 29          C-----                                LIS19290
39 30          CCOMMON IP, JP, IBR, IPP, JPP, ATENDA, LIS19300
39 31          C          NFREEP, NFREEB, LASTR, ARG, ARG2, ARG3, LIS19310
39 32          C          ARG4, MODE, ALIST, SUBR0, SUBR1, SUBR2, LIS19320
39 33          C          SUBR3, A2, A8, A9, LAMBDA, LABEL, LIS19330
39 34          C          FUNARG, NIL, EXPR, FEXPR, SUBR, FSUBR, LIS19340
39 35          C          T, SYS, RDPNT, PRTPNT, BSTACK(5), LIS19350
39 36          C          DREG(24)                     LIS19360
39 37          COMMON /CONST/ NATCM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,LUNUTL LIS19370
39 38          COMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC, LIS19380
39 39          C IPLUS,IMINUS,IFIG(10),ATEND,ALLSP      LIS19390
39 40          CCOMMON /CARSTR/ NILCAR(2),CAR(10)      LIS19400
39 41          COMMON /CDRSTR/ NILCDR(2),CDR(10)      LIS19410
39 42          CCOMMON/ATOM1/ FNAME(4008)             LIS19420
39 43          CCOMMON/ATOM2/ FNAME1,PNAME2           LIS19430
39 44          LCGICAL*1 PNAME                         LIS19440
39 45          CCOMMON /STKSTR/ STACK(10)             LIS19450
39 46          INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT, LIS19460
39 47          I          ALLSP, RBRACK, UNDEF, ATEND, ARG, ARG2, LIS19470
39 48          I          ARG3, ARG4, ALIST, SUBR0, SUBR1, SUBR2, LIS19480
39 49          I          SUBR3, A2, A8, A9, FUNARG, EXPR, LIS19490
39 50          I          FEXPR, SUBR, FSUBR, T, SYS, RDPNT, LIS19500
40 1          I          PRTPNT, ATENDA, BSTACK, DREG LIS19510
40 2          INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR LIS19520
40 3          COMMON AND INTEGER DECLARATIONS ENDED LIS19530
40 4          C-----                                LIS19540
40 5          C                                       LIS19550
40 6          C0000 CCNS 000000000000               LIS19560
40 7          2 IF(CDR(NFREEP)) 5,10,5              LIS19570
40 8          5 ISL=CDR(NFREEP)                     LIS19580
40 9          CAR(NFREEP)=I1                          LIS19590
40 10         CDR(NFREEP)=I2                         LIS19600
40 11         CONS=NFREEP                           LIS19610
40 12         NFREEP=ISL                             LIS19620
40 13         RETURN                                 LIS19630
40 14         C                                       LIS19640
40 15         C0000 FREE LIST EMPTY0000000000000000000000000000000000000000 LIS19650
40 16         C0000 DO GARB,COLL. ON I1,I2,ARG,ARG2,ARG3,ARG4,LASTR,STACK LIS19660
40 17         10 CALL GARB(I1)                       LIS19670
40 18         CALL GARB(I2)                           LIS19680
40 19         CALL GARB(ARG)                           LIS19690
40 20         CALL GARB(ARG2)                           LIS19700
40 21         CALL GARB(ARG3)                           LIS19710
40 22         CALL GARB(ARG4)                           LIS19720
40 23         CALL GARB(LASTR)                           LIS19730
40 24         DO 20 I=JP,NSTACK                          LIS19740
40 25         ISTACK=STACK(I)                           LIS19750
40 26         20 CALL GARB(ISTACK)                       LIS19760
40 27         DO 25 I=1,NATOMP                           LIS19770
40 28         ICAR=CAR(I)                               LIS19780

```

```

40 29          ICDR=CDR(I)
40 30          CALL GARB(ICAR)
40 31          25 CALL GARB(ICDR)
40 32          CALL MAKFRE
40 33          IF(DREG(1)-T) 2,30,2
40 34          30 NR=LISTNB(NFREEP)
40 35          WRITE(LUNUT,100) NR
40 36          100 FORMAT (22H GARB.CCLL.  FREEL IST=IS)
40 37          GO TO 2
40 38          END
40 39          FUNCTION LISTNB(IJK)
40 40          COMMON AND INTEGER DECLARATIONS
40 41          C-----
40 42          COMMON IP, JP, IBR, IPP, JPP, ATENDA,
40 43          C          NFREEP, NFREEB, LASTR, ARG, ARG2, ARG3,
40 44          C          ARG4, MODE, ALIST, SUBR0, SUBR1, SUBR2,
40 45          C          SUBR3, A2, A8, A9, LAMBDA, LABEL,
40 46          C          FUNARG, NIL, EXPR, FEXPR, SUBR, FSUBR,
40 47          C          T, SYS, RDPNT, RPTPNT, BSTACK(5),
40 48          C          DREG(24)
40 49          COMMON /CONST/ NATOM,NFREEE,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,LUNUT
40 50          COMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC,
41 1          C IPLUS,IMINUS,IFIG(10),ATEND,ALLSP
41 2          COMMON /CARSTR/ NILCAR(2),CAR(10)
41 3          COMMON /CDRSTR/ NILCDR(2),CDR(10)
41 4          COMMON/ATOM1/ PNAME(4008)
41 5          COMMON/ATOM2/ PNAME1,PNAME2
41 6          LOGICAL*1 PNAME
41 7          COMMON /STKSTR/ STACK(10)
41 8          INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT,
41 9          I          ALLSP, RBRACK, UNDEF, ATEND, ARG, ARG2,
41 10          I          ARG3, ARG4, ALIST, SUBR0, SUBR1, SUBR2,
41 11          I          SUBR3, A2, A8, A9, FUNARG, EXPR,
41 12          I          FEXPR, SUBR, FSUBR, T, SYS, RDPNT,
41 13          I          RPTPNT, ATENDA, BSTACK, DREG
41 14          INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR
41 15          COMMON AND INTEGER DECLARATIONS ENDED
41 16          C-----
41 17          C
41 18          C0000 COUNTS THE ELEMENT IN A ONE-WAY LIST
41 19          LISTST=IJK
41 20          I=0
41 21          IF(LISTST) 1,10,1
41 22          1 I=1
41 23          2 IF(CDR(LISTST)) 5,10,5
41 24          5 I=I+1
41 25          LISTST=CDR(LISTST)
41 26          GO TO 2
41 27          10 LISTNB=I
41 28          RETURN
41 29          END
41 30          FUNCTION IREAD(IDUMMY)
41 31          COMMON AND INTEGER DECLARATIONS
41 32          C-----
41 33          COMMON IP, JP, IBR, IPP, JPP, ATENDA,
41 34          C          NFREEP, NFREEB, LASTR, ARG, ARG2, ARG3,
41 35          C          ARG4, MODE, ALIST, SUBR0, SUBR1, SUBR2,
41 36          C          SUBR3, A2, A8, A9, LAMBDA, LABEL,
41 37          C          FUNARG, NIL, EXPR, FEXPR, SUBR, FSUBR,
41 38          C          T, SYS, RDPNT, RPTPNT, BSTACK(5),

```

```

41 39      C      DREG(24)
41 40      COMMON /CONST/ NATOM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,LUNUTL LIS20390
41 41      CCOMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC, LIS20400
41 42      C IPLUS,IMINUS,IFIG(10),ATEND,ALLSP LIS20420
41 43      CCOMMON /CARSTR/ NILCAR(2),CAR(10) LIS20430
41 44      CCOMMON /CDRSTR/ NILCDR(2),CDR(10) LIS20440
41 45      CCOMMON/ATOM1/ PNAME(4008) LIS20450
41 46      CCOMMON/ATOM2/ PNAME1,PNAME2 LIS20460
41 47      LOGICAL*1 PNAME LIS20470
41 48      CCOMMON /STKSTR/ STACK(10) LIS20480
41 49      INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT, LIS20490
41 50      I ALLSP, RBRACK, UNDEF, ATEND, ARG, ARG2, LIS20500
42 1      I ARG3, ARG4, ALIST, SUBR0, SUBR1, SUBR2, LIS20510
42 2      I SUBR3, A2, A8, A9, FUNARG, EXPR, LIS20520
42 3      I FEXPR, SUBR, FSUBR, T, SYS, RDPNT, LIS20530
42 4      I PRFNT, ATENCA, BSTACK, DREG LIS20540
42 5      INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR LIS20550
42 6      COMMON AND INTEGER DECLARATIONS ENDED LIS20560
42 7      C----- LIS20570
42 8      INTEGER CONS,RATOM LIS20580
42 9      C LIS20590
42 10     C0000IREAD IS A NO RECURSIVE READ WHITCH READ ONE S-EXPRESSI LIS20600
42 11     C0000THE VALUE IS IREAD LIS20610
42 12     C0000LASTR SAVES THE READLIST IN CASE OF GARB.COLL. LIS20620
42 13     C0000 FIRST CALL IN ORDER TO GIVE INITIAL VALUES LIS20630
42 14     NAT=0 LIS20640
42 15     NAT1=0 LIS20650
42 16     10 IDUM=RATOM(-1) LIS20660
42 17     ITYP=RATOM(NAT) LIS20670
42 18     IF(ITYP-3) 21,20,21 LIS20680
42 19     20 IF(IPROCI-9) 26,25,26 LIS20690
42 20     25 IPRCCI=1 LIS20700
42 21     IREAD=-1 LIS20710
42 22     ARG=NAT LIS20720
42 23     RETURN LIS20730
42 24     26 IREAD=NAT LIS20740
42 25     RETURN LIS20750
42 26     21 IF(ITYP-5) 22,500,22 LIS20760
42 27     22 LAST=CONS(NIL,NIL) LIS20770
42 28     LASTR=LAST LIS20780
42 29     C0000 SAVE LASTR IN CASE OF GARB.COLL. LIS20790
42 30     GO TO 51 LIS20800
42 31     50 ITYP=ITYP1 LIS20810
42 32     NAT=NAT1 LIS20820
42 33     51 ITYP1=RATOM(NAT1) LIS20830
42 34     GO TO (100,200,300,400,500),ITYP LIS20840
42 35     C ( ) AT ERR ( LIS20850
42 36     C0000 ( 0000C00000000000000000000000000000 LIS20860
42 37     C FIRST CHECK IF NEXT=} LIS20870
42 38     100 IF(ITYP1-2) 130,110,130 LIS20880
42 39     110 IF(CDR(LAST)) 125,120,125 LIS20890
42 40     120 IREAD=NIL LIS20900
42 41     GO TO 1000 LIS20910
42 42     125 NAT=NIL LIS20920
42 43     ITYP1=RATOM(NAT1) LIS20930
42 44     GO TO 300 LIS20940
42 45     C GO TO ATCM-ROUTINE LIS20950
42 46     C NOW DO THE ( ROUTINE LIS20960
42 47     130 NEW=CONS(NIL,LAST) LIS20970
42 48     CAR(LAST)=NEW LIS20980

```

```

42 49 LAST=NEW LIS20990
42 50 GO TO 50 LIS21000
43 1 C0000 ATCM C000C000000000000000000000000000 LIS21010
43 2 300 CAR(LAST)=NAT LIS21020
43 3 305 IF(ITYP1-2) 330,310,330 LIS21030
43 4 310 NEW=CDR(LAST) LIS21040
43 5 CDR(LAST)=NAT1 LIS21050
43 6 LAST=NEW LIS21060
43 7 IF(CDR(LAST)) 320,999,320 LIS21070
43 8 320 ITYP1=RATCM(NAT1) LIS21080
43 9 GO TO 305 LIS21090
43 10 C *** ( OR ATCM AFTER ATCM LIS21100
43 11 330 ICDR=CDR(LAST) LIS21110
43 12 NEW=CONS(NIL,ICDR) LIS21120
43 13 CDR(LAST)=NEW LIS21130
43 14 LAST=NEW LIS21140
43 15 GO TO 50 LIS21150
43 16 C@@@@ RETURN LIS21160
43 17 999 IREAD=CAR(LAST) LIS21170
43 18 1000 LASTR=NIL LIS21180
43 19 C0000 NO USE OF LASTR ANY LONGER LIS21190
43 20 IF(IPROCI-9) 1020,1010,1020 LIS21200
43 21 1010 ARG=IREAD LIS21210
43 22 IREAD=-1 LIS21220
43 23 IPROCI=1 LIS21230
43 24 1020 RETURN LIS21240
43 25 200 WRITE(LUNUT,123) LIS21250
43 26 123 FORMAT (38H --- READ ERROR. S-EXPR. BEGINS WITH )) LIS21260
43 27 400 CALL SHIFT(123456) LIS21270
43 28 NAT=0 LIS21280
43 29 GO TO 10 LIS21290
43 30 500 IPROCI=9 LIS21300
43 31 GO TO 10 LIS21310
43 32 END LIS21320
43 33 FUNCTION MATOM(FN1,PN2) LIS21330
43 34 INTEGER PN2 LIS21340
43 35 LOGICAL*1 PN1(1) LIS21350
43 36 CCMCN AND INTEGER DECLARATIONS LIS21360
43 37 C----- LIS21370
43 38 CCMCN IP, JP, IBR, IPP, JPP, ATENDA, LIS21380
43 39 C NFREEP, NFREEB, LASTR, ARG, ARG2, ARG3, LIS21390
43 40 C ARG4, MODE, ALIST, SUBRO, SUBR1, SUBR2, LIS21400
43 41 C SUBR3, A2, A8, A9, LAMBDA, LABEL, LIS21410
43 42 C FUNARG, NIL, EXPR, FEXPR, SUBR, FSUBR, LIS21420
43 43 C T, SYS, RDPNT, PRTPNT, BSTACK(5), LIS21430
43 44 C DREG(24) LIS21440
43 45 COMMON /CONST/ NATOM,NFREEP,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,LUNUT LIS21450
43 46 COMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC, LIS21460
43 47 C IPLUS,IMINUS,IFIG(10),ATEND,ALLSP LIS21470
43 48 CCMCN /CARSTR/ NILCAR(2),CAR(10) LIS21480
43 49 CCMCN /CDRSTR/ NILCDR(2),CDR(10) LIS21490
43 50 CCMCN/ATOM1/ PNAME(4008) LIS21500
44 1 CCMCN/ATOM2/ PNAME1,PNAME2 LIS21510
44 2 LCGICAL*1 PNAME LIS21520
44 3 CCMCN /STKSTR/ STACK(10) LIS21530
44 4 INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT, LIS21540
44 5 I ALLSP, RBRACK, UNDEF, ATEND, ARG, ARG2, LIS21550
44 6 I ARG3, ARG4, ALIST, SUBRO, SUBR1, SUBR2, LIS21560
44 7 I SUBR3, A2, A8, A9, FUNARG, EXPR, LIS21570
44 8 I FEXPR, SUBR, FSUBR, T, SYS, RDPNT, LIS21580

```

```

44 9          I          PRTFNT,  ATENDA,  BSTACK,  DREG          LIS21590
44 10         INTEGER PNAMEP,PNAME1          LIS21600
44 11         INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR          LIS21610
44 12         CCMCN AND INTEGER DECLARATIONS ENDED          LIS21620
44 13         C-----          LIS21630
44 14         C          LIS21640
44 15         C0000 MATOM CHECKS IF THE ATOM EXISTS.          LIS21650
44 16         C0000 IN THAT CASE RETURN WITH THAT VALUE, ELSE          LIS21660
44 17         C0000 CREATE A NEW ATOM          LIS21670
44 18         MATCHM=PNAME1(FN1,PN2)          LIS21680
44 19         IF(MATCHM.GE.0)RETURN          LIS21690
44 20         MATCHM=PNAME2          LIS21700
44 21         PNAME2=FN2          LIS21710
44 22         C0000 ATOM NOT FOUND. MAKE NEW ONE.          LIS21720
44 23         40 IF(MATOM-NATOM) 60,60,50          LIS21730
44 24         50 WRITE(LUNUT,501) NATOM          LIS21740
44 25         501 FCRMAT (26H OBJECT STACK IS EXCEEDINGI5)          LIS21750
44 26         MATCHM=NIL          LIS21760
44 27         RETURN          LIS21770
44 28         60 NATCMP=MATCHM          LIS21780
44 29         CAR(MATCHM)= -FNAME1          LIS21790
44 30         CAR(MATOM+1)= -(PNAME1+PNAME2+1)          LIS21800
44 31         IF(PNAME1+PNAME2+1 .GT. 8192) GOTO 90          LIS21810
44 32         CDR(MATCHM)=NIL          LIS21820
44 33         CALL MVC(FNAME(PNAME1),FN1,PNAME2,PNAME(PNAME1+PNAME2),'??',2)          LIS21830
44 34         RETURN          LIS21840
44 35         90 WRITE(LUNUT,91)(PN1(I),I=1,PN2)          LIS21850
44 36         91 FORMAT(* PNAME LIST EXCEEDED AT ',80A1)          LIS21860
44 37         CALL SVC(15)          LIS21870
44 38         END          LIS21880
44 39         INTEGER FUNCTION RATOM(IATOM)          LIS21890
44 40         COMMON AND INTEGER DECLARATIONS          LIS21900
44 41         C-----          LIS21910
44 42         COMMON IP, JP, IBR, IPP, JPP, ATENDA,          LIS21920
44 43         C NFREEP, NFREEB, LASTR, ARG, ARG2, ARG3,          LIS21930
44 44         C ARG4, MODE, ALIST, SUBR0, SUBR1, SUBR2,          LIS21940
44 45         C SUBR3, A2, A8, A9, LAMBDA, LABEL,          LIS21950
44 46         C FUNARG, NIL, EXPR, FEXPR, SUBR, FSUBR,          LIS21960
44 47         C T, SYS, RDPNT, PRTPNT, BSTACK(5),          LIS21970
44 48         C DREG(24)          LIS21980
44 49         COMMON /CCNST/ NATOM,NFREET,NSTACK,NATCMP,NUMADD,UNDEF,LUNIN,LUNUT          LIS21990
44 50         COMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC,          LIS22000
45 1         C IPLUS,IMINUS,IFIG(10),ATEND,ALLSP          LIS22010
45 2         CCMCN /CARSTR/ NILCAR(2),CAR(10)          LIS22020
45 3         COMMON /CDRSTR/ NILCDR(2),CDR(10)          LIS22030
45 4         COMMON/ATOM1/ PNAME(4008)          LIS22040
45 5         CCMCN/ATOM2/ PNAME1,PNAME2          LIS22050
45 6         LOGICAL*1 PNAME          LIS22060
45 7         CCMCN /STKSTR/ STACK(10)          LIS22070
45 8         INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT,          LIS22080
45 9         I ALLSP, RBRACK, UNDEF, ATEND, ARG, ARG2,          LIS22090
45 10        I ARG3, ARG4, ALIST, SUBR0, SUBR1, SUBR2,          LIS22100
45 11        I SUBR3, A2, A8, A9, FUNARG, EXPR,          LIS22110
45 12        I FEXPR, SUBR, FSUBR, T, SYS, RDPNT,          LIS22120
45 13        I PRTPNT, ATENDA, BSTACK, DREG          LIS22130
45 14        INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR          LIS22140
45 15        COMMON AND INTEGER DECLARATIONS ENDED          LIS22150
45 16        C-----          LIS22160
45 17        INTEGER BRDUM(1),SBRACK(5),RBRIND          LIS22170
45 18        IDUM=0          LIS22180

```

45	19	IF(IATCM) 1,2,2	LIS22190
45	20	C0000 FIRST GIVE INITIAL-VALUE IN JATOM	LIS22200
45	21	1 CALL JATOM(1,IATOM)	LIS22210
45	22	IBRACK=0	LIS22220
45	23	RBRIND=1	LIS22230
45	24	BRDUM(1)=0	LIS22240
45	25	DC 123 I=1,5	LIS22250
45	26	123 SBRACK(I)=0	LIS22260
45	27	RATOM= IATOM	LIS22270
45	28	RETURN	LIS22280
45	29	2 GO TO (10,3),RBRIND	LIS22290
45	30	C RBRACK HAS BEEN READ. RETURN)	LIS22300
45	31	3 IF(SBRACK(IBRACK))5,5,4	LIS22310
45	32	4 RATOM=2	LIS22320
45	33	SBRACK(IBRACK)=SBRACK(IBRACK)-1	LIS22330
45	34	IATOM=NIL	LIS22340
45	35	RETURN	LIS22350
45	36	5 RBRIND=1	LIS22360
45	37	IF(IBRACK) 10,10,6	LIS22370
45	38	6 IBRACK=IBRACK-1	LIS22380
45	39	GO TO 10	LIS22390
45	40	10 CALL JATOM(ITYP,IATOM)	LIS22400
45	41	GC TO (20,30,40,11,50,60,70),ITYP	LIS22410
45	42	C A () < > %	LIS22420
45	43	11 CALL JATOM(ITYP,IATCM)	LIS22430
45	44	IF(ITYP-1) 12,13,12	LIS22440
45	45	12 WRITE(LUNUT,100)	LIS22450
45	46	100 FORMAT (41H --- READ ERROR . NOT FOLLOWED BY ATOM)	LIS22460
45	47	121 CALL SHIFT(123456)	LIS22470
45	48	RATOM=4	LIS22480
45	49	RETURN	LIS22490
45	50	13 CALL JATOM(ITYP,IDUM)	LIS22500
46	1	IF(ITYP-3) 14,15,14	LIS22510
46	2	14 WRITE(LUNUT,101)	LIS22520
46	3	101 FORMAT (42H --- READ ERROR .ATOM) NOT FOLLOWED BY))	LIS22530
46	4	GO TO 121	LIS22540
46	5	15 RATOM=2	LIS22550
46	6	RETURN	LIS22560
46	7	20 RATOM=3	LIS22570
46	8	RETURN	LIS22580
46	9	30 SBRACK(IBRACK)=SBRACK(IBRACK)+1	LIS22590
46	10	RATOM=1	LIS22600
46	11	RETURN	LIS22610
46	12	40 SBRACK(IBRACK)=SBRACK(IBRACK)-1	LIS22620
46	13	RATCM=2	LIS22630
46	14	IATCM=NIL	LIS22640
46	15	RETURN	LIS22650
46	16	50 IF(IBRACK-5) 51,30,30	LIS22660
46	17	51 IBRACK=IBRACK+1	LIS22670
46	18	GO TO 30	LIS22680
46	19	60 RBRIND=2	LIS22690
46	20	GC TO 3	LIS22700
46	21	70 RATOM=5	LIS22710
46	22	RETURN	LIS22720
46	23	END	LIS22730
46	24	SUBROUTINE SBYT(L,IB,K)	LIS22740
46	25	C USED BY JATCM TO PUT ONE CHARACTER IN IB	LIS22750
46	26	LOGICAL*1 IB(1),L(1)	LIS22760
46	27	K=K+1	LIS22770
46	28	IB(K)=L(1)	LIS22780


```

46 29 RETURN LIS22790
46 30 END LIS22800
46 31 SUBROUTINE SHIFT(IC) LIS22810
46 32 COMMON AND INTEGER DECLARATIONS LIS22820
46 33 C----- LIS22830
46 34 COMMON IP, JP, IBR, IPP, JPP, ATENDA, LIS22840
46 35 C NFREEP, NFREEB, LASTR, ARG, ARG2, ARG3, LIS22850
46 36 C ARG4, MODE, ALIST, SUBR0, SUBR1, SUBR2, LIS22860
46 37 C SUBR3, A2, A8, A9, LAMBDA, LABEL, LIS22870
46 38 C FUNARG, NIL, EXPR, FEXPR, SUBR, FSUBR, LIS22880
46 39 C T, SYS, RDPNT, PRTPNT, BSTACK(5), LIS22890
46 40 C DREG(24) LIS22900
46 41 COMMON /CONST/ NATCM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,LUNUT LIS22910
46 42 COMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC, LIS22920
46 43 C IPLUS,IMINUS,IFIG(10),ATEND,ALLSP LIS22930
46 44 COMMON /CARSTR/ NILCAR(2),CAR(10) LIS22940
46 45 COMMON /CDRSTR/ NILCDR(2),CDR(10) LIS22950
46 46 COMMON/ATCM1/ FNAME(4008) LIS22960
46 47 COMMON/ATCM2/ FNAME1,PNAME2 LIS22970
46 48 LOGICAL*1 PNAME LIS22980
46 49 COMMON /STKSTR/ STACK(10) LIS22990
46 50 INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT, LIS23000
47 1 I ALLSP, RERACK, UNDEF, ATEND, ARG, ARG2, LIS23010
47 2 I ARG3, ARG4, ALIST, SUBR0, SUBR1, SUBR2, LIS23020
47 3 I SUBR3, A2, A8, A9, FUNARG, EXPR, LIS23030
47 4 I FEXPR, SUBR, FSUBR, T, SYS, RDPNT, LIS23040
47 5 I PRTFNT, ATENDA, BSTACK, DREG LIS23050
47 6 INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR LIS23060
47 7 COMMON AND INTEGER DECLARATIONS ENDED LIS23070
47 8 C----- LIS23080
47 9 DIMENSION LIN(72) LIS23090
47 10 EQUIVALENCE (K,RDPNT) LIS23100
47 11 IF(IC-123456) 6,5,6 LIS23110
47 12 5 K=1000 LIS23120
47 13 RETURN LIS23130
47 14 6 IF(K-72) 10,10,20 LIS23140
47 15 20 READ(LUNIN,100,END=30) LIN LIS23150
47 16 100 FORMAT(72A1) LIS23160
47 17 K=1 LIS23170
47 18 10 IC=LIN(K) LIS23180
47 19 K=K+1 LIS23190
47 20 RETURN LIS23200
47 21 30 CALL SVC(14) LIS23210
47 22 END LIS23220
47 23 SUBROUTINE GARB(IJK) LIS23230
47 24 COMMON AND INTEGER DECLARATIONS LIS23240
47 25 C----- LIS23250
47 26 COMMON IP, JP, IBR, IPP, JPP, ATENDA, LIS23260
47 27 C NFREEP, NFREEB, LASTR, ARG, ARG2, ARG3, LIS23270
47 28 C ARG4, MODE, ALIST, SUBR0, SUBR1, SUBR2, LIS23280
47 29 C SUBR3, A2, A8, A9, LAMBDA, LABEL, LIS23290
47 30 C FUNARG, NIL, EXPR, FEXPR, SUBR, FSUBR, LIS23300
47 31 C T, SYS, RDPNT, PRTPNT, BSTACK(5), LIS23310
47 32 C DREG(24) LIS23320
47 33 COMMON /CONST/ NATOM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN, LIS23330
47 34 C LUNUT,IBIG,IRANGE LIS23340
47 35 COMMON /SYMBOL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC, LIS23350
47 36 C IPLUS,IMINUS,IFIG(10),ATEND,ALLSP LIS23360
47 37 COMMON /CARSTR/ NILCAR(2),CAR(10) LIS23370
47 38 COMMON /CDRSTR/ NILCDR(2),CDR(10) LIS23380

```

47	39	CCOMON/ATCM1/ PNAME(4008)	LIS23390
47	40	CCOMON/ATCM2/ PNAME1,PNAME2	LIS23400
47	41	LOGICAL*1 PNAME	LIS23410
47	42	CCOMON /STKSTR/ STACK(10)	LIS23420
47	43	INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT,	LIS23430
47	44	I ALLSP, RBRACK, UNDEF, ATEND, ARG, ARG2,	LIS23440
47	45	I ARG3, ARG4, ALIST, SUBRO, SUBR1, SUBR2,	LIS23450
47	46	I SUBR3, A2, A8, A9, FUNARG, EXPR,	LIS23460
47	47	I FEXPR, SUBR, FSUBR, T, SYS, RDPNT,	LIS23470
47	48	I PRFTNT, ATENCA, BSTACK, DREG	LIS23480
47	49	INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR	LIS23490
47	50	COMMON AND INTEGER DECLARATIONS ENDED	LIS23500
48	1	C-----	LIS23510
48	2	C0000 IBIG MUST FIT IN CAR,CDR	LIS23520
48	3	C	LIS23530
48	4	C A NO-RECURSIVE GARBAGE-COLLECTOR WHITCH USES THE ALGOR	LIS23540
48	5	C DESCRIBED IN CACM AUG 67 (NR 8)	LIS23550
48	6	C ILEFT POINTS TO THE LEFT CELL	LIS23560
48	7	C I POINTS TO THIS CELL	LIS23570
48	8	C IRIGHTPOINTS TO THE RIGHT CELL	LIS23580
48	9	C	LIS23590
48	10	I=IJK	LIS23600
48	11	ILEFT= IBIG	LIS23610
48	12	C00000 FIRST TEST IF I POINTS TO A LIST IN FREE-MEMORY	LIS23620
48	13	IF(NFREET-I) 95,2,2	LIS23630
48	14	2 IF(NATCM-I) 5,95,95	LIS23640
48	15	C0000 FORWARD SCAN000000000000000000000000	LIS23650
48	16	5 IF(CAR(I)) 20,9,9	LIS23660
48	17	9 IF(CDR(I)) 20,10,10	LIS23670
48	18	10 CAR(I)=-CAR(I)-1	LIS23680
48	19	IRIGHT=CDR(I)	LIS23690
48	20	CDR(I)=ILEFT	LIS23700
48	21	IF(IRIGHT-NATCM) 50,50,15	LIS23710
48	22	15 IF(IRIGHT-NFREET) 16,16,50	LIS23720
48	23	16 ILEFT=I	LIS23730
48	24	I=IRIGHT	LIS23740
48	25	GO TO 5	LIS23750
48	26	C0000 REVERSE SCAN 000000000000000000000000	LIS23760
48	27	20 I=ILEFT	LIS23770
48	28	50 IF(I- IBIG) 55,90,55	LIS23780
48	29	55 IF(CAR(I)- IBIG) 60,91,60	LIS23790
48	30	60 IF(CDR(I)) 75,70,70	LIS23800
48	31	C0000 NO MARK FOR BRANCH-POINT	LIS23810
48	32	70 ILEFT=CDR(I)	LIS23820
48	33	CDR(I)=IRIGHT	LIS23830
48	34	GO TO 80	LIS23840
48	35	C0000 MARK FOR BRANCH-POINT	LIS23850
48	36	75 ISL=I	LIS23860
48	37	I=CAR(I)	LIS23870
48	38	CAR(ISL)=-IRIGHT-1	LIS23880
48	39	CDR(ISL)=-CDR(ISL)-1	LIS23890
48	40	IRIGHT=ISL	LIS23900
48	41	GO TO 50	LIS23910
48	42	C0000 OBS. CAR(I) IS NEG. HERE	LIS23920
48	43	C0000 TEST FOR SUBLIST	LIS23930
48	44	80 IF(NATCM+CAR(I)+1) 81,85,85	LIS23940
48	45	81 IF(-1-CAR(I)-NFREET) 82,82,85	LIS23950
48	46	C0000 CAR(I) POINTS TO A SUBLIST	LIS23960
48	47	82 IRIGHT=-CAR(I)-1	LIS23970
48	48	CAR(I)=ILEFT	LIS23980

48	49	CDR(I)=-CDR(I)-1	LIS23990
48	50	GO TO 16	LIS24000
49	1	C0000 CAR(I) DOES NOT POINT TO A LIST	LIS24010
49	2	C0000 GO ON WITH BACK-UP	LIS24020
49	3	85 IRIGHT=I	LIS24030
49	4	I=ILEFT	LIS24040
49	5	GO TO 50	LIS24050
49	6	90 I=IRIGHT	LIS24060
49	7	GO TO 95	LIS24070
49	8	91 CAR(I)=-IRIGHT-1	LIS24080
49	9	CDR(I)=-CDR(I)-1	LIS24090
49	10	95 RETURN	LIS24100
49	11	END	LIS24110
49	12	SUBROUTINE PUTCH(OUTWOR,INWORD,NUMB)	LIS24120
49	13	INTEGER OUTWOR, STOREI	LIS24130
49	14	LOGICAL*1 INWORD(4),STOREL(4)	LIS24140
49	15	EQUIVALENCE (STOREI,STOREL(1))	LIS24150
49	16	STOREI = OUTWOR	LIS24160
49	17	STOREL(NUMB)= INWORD(1)	LIS24170
49	18	OUTWOR = STOREI	LIS24180
49	19	RETURN	LIS24190
49	20	END	LIS24200
49	21	SUBROUTINE GETCH(INWORD,OUTWOR,NUMB)	LIS24210
49	22	INTEGER OUTWOR, BLANK/' */	LIS24220
49	23	LOGICAL*1 INWORD(4),STOREL(4)	LIS24230
49	24	EQUIVALENCE (BLANK, STOREL(1))	LIS24240
49	25	STOREL(1) = INWORD(NUMB)	LIS24250
49	26	OUTWOR = BLANK	LIS24260
49	27	RETURN	LIS24270
49	28	END	LIS24280
49	29	INTEGER FUNCTION ARITH(TYPE,ARGU1,ARGU2,FCN)	LIS24290
49	30	C TYPE 1 - ADD, 2- SUBTRACT, 3 - MULTIPLY, 4 - DIVIDE	LIS24300
49	31	COMMON /CONST/ NATOM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN,	LIS24310
49	32	C LUNUT,IBIG,IRANGE	LIS24320
49	33	COMMON/ATCM1/ PNAME(4008)	LIS24330
49	34	COMMON/ATOM2/ PNAME1,PNAME2	LIS24340
49	35	LOGICAL*1 PNAME	LIS24350
49	36	INTEGER TYPE,ARGU1,ARGU2,FCN,UNDEF,PNAME1,PNAME2	LIS24360
49	37	INTEGER PNAMEP,PNAMEN	LIS24370
49	38	IF (ARGU1-NFREET) 600,600,10	LIS24380
49	39	10 J = ARGU1-NUMADD	LIS24390
49	40	IF (ARGU2-NFREET) 600,600,20	LIS24400
49	41	20 K= ARGU2-NUMADD	LIS24410
49	42	GO TO (100,200,300,400), TYPE	LIS24420
49	43	C ADD ARGU1,ARGU2	LIS24430
49	44	100 J=J+K	LIS24440
49	45	GO TO 500	LIS24450
49	46	C SUBTRACT ARGU1,ARGU2	LIS24460
49	47	200 J= J-K	LIS24470
49	48	GO TO 500	LIS24480
49	49	C MULTIPLY ARGU1,ARGU2	LIS24490
49	50	300 J=J*K	LIS24500
50	1	GO TO 500	LIS24510
50	2	C DIVIDE ARGU1,ARGU2	LIS24520
50	3	400 IF (K) 410,570,410	LIS24530
50	4	410 J=J/K	LIS24540
50	5	C CHECK RESULT	LIS24550
50	6	500 IF (J-IRANGE) 510,510,530	LIS24560
50	7	510 IF (J+IRANGE) 550,520,520	LIS24570
50	8	520 ARITH = J+NUMADD	LIS24580

50	9	RETURN	LIS24590	
50	10	530 J=IRANGE	LIS24600	
50	11	534 CALL ERROR(9,FCN)	LIS24610	
50	12	WRITE(3,540)J	LIS24620	
50	13	540 FORMAT(2X,'VALUE SET TO ',I6)	LIS24630	
50	14	GO TO 520	LIS24640	
50	15	550 J = -IRANGE	LIS24650	
50	16	GO TO 534	LIS24660	
50	17	570 CALL EFFOR(10,FCN)	LIS24670	
50	18	WRITE(3,540)IRANGE	LIS24680	
50	19	J=IRANGE	LIS24690	
50	20	GO TO 520	LIS24700	
50	21	600 CALL ERROR(8,FCN)	LIS24710	
50	22	RETURN	LIS24720	
50	23	END	LIS24730	
50	24	INTEGER FUNCTION PNAMEP(PT)	LIS24740	
50	25	COMMON/ATCM1/ PNAME(1)	LIS24750	
50	26	COMMON/ATCM2/ FNAME1,PNAME2	LIS24760	
50	27	COMMON/CARSTR/NILCAR(2),CAR(10)	LIS24770	
50	28	INTEGER PNAME1,FNAME2,FNAMEN,PT,PNL,CLC,NILCAR*2,CAR*2	LIS24780	
50	29	LOGICAL*1 PNAME,NE,PN(1),PUNCT/'?'/	LIS24790	
50	30		LIS24800	
50	31	PNAME1=IAES(CAR(PT)+0)	LIS24810	
50	32	PNAME2=IAES(CAR(PT+1)+0)-PNAME1-1	LIS24820	
50	33	PNAMEP=FNAME1	LIS24830	
50	34	RETURN	LIS24840	
50	35		LIS24850	
50	36	ENTRY FNAMEN(PN,PNL)	LIS24860	
50	37	PNAMEN=-1	LIS24870	
50	38	J=0	LIS24880	
50	39	25 PNAMEN=FNAMEN+1	LIS24890	
50	40	PNAME1=J+1	LIS24900	
50	41	30 J=J+1	LIS24910	
50	42	IF(NE(FNAME(J),PUNCT)) GO TO 30	LIS24920	
50	43	PNAME2=J-PNAME1	LIS24930	
50	44	IF(PNAME2.LE.0) GO TO 40	LIS24940	
50	45	IF(PNAME2.NE.PNL) GO TO 25	LIS24950	
50	46	IF(CLC(FNAME(FNAME1),PN,PNAME2).NE.0) GO TO 25	LIS24960	
50	47	RETURN	LIS24970	
50	48	40 PNAME2=FNAMEN	LIS24980	
50	49	PNAMEN=-1	LIS24990	
50	50	RETURN	LIS25000	
51	1	END	LIS25010	
51	2	SUBROUTINE ERROR(N,ARG)	LIS25020	
51	3	COMMON/ATOM1/ PNAME(1)	LIS25030	
51	4	COMMON/ATCM2/ FNAME1,PNAME2	LIS25040	
51	5	INTEGER PNAME1,PNAME2,FNAMEP,ARG	LIS25050	
51	6	INTEGER ACODE,CVB	LIS25060	
51	7	LOGICAL*1 PNAME	LIS25070	
51	8	REAL*8 ACTICN(2)/* RESTART', ' BREAK */	LIS25080	
51	9	LOGICAL*1 MSG(40,10)	LIS25090	
51	10	LOGICAL*1 MSG1(40,5)/* 1 UNDEFINED FUNCTION	1 UNLIS25100	
51	11	1 UNDEFINED VARIABLE	2 FIRST ARG OF SET/SETQ NOT ONLIS25110	
51	12	2 A-LIST	2 ARG OF GO TO NOT DEFINED	1 INCORRECT NUMBLIS25120
51	13	3ER OF ARGUMENTS FOR	*/	LIS25130
51	14	LOGICAL*1 MSG2(40,5)/* 1 ARGUMENT OF IMPROPER FORM FOR	1 CALIS25140	
51	15	1R OF AN ATOM ATEMPTED IN FUNCTION	1 NON-INTEGGER ARGUMENT IN	LIS25150
51	16	2	0 INTEGER OVERFLOW IN	0 DIVISION BY ZELIS25160
51	17	3RC IN	*/	LIS25170
51	18	EQUIVALENCE (MSG(1,1),MSG1),(MSG(1,6),MSG2)	LIS25180	

```

51 19          WRITE(3,11) (MSG(I,N), I=3,40), (PNAME(PNAMEP(ARG)+I-1), I=1,PNAME2) LIS25190
51 20          11 FORMAT(1X,38A1,1X,80A1) LIS25200
51 21             ACODE=CVB(MSG(1,N),1) LIS25210
51 22             IF(ACODE.GT.0)WRITE(3,12)ACTION(ACODE) LIS25220
51 23          12 FORMAT(1X,A8) LIS25230
51 24             RETURN 1 LIS25240
51 25             END LIS25250
51 26             SUBROUTINE JATCM(IRET,IATOM) LIS25260
51 27 COMMON AND INTEGER DECLARATIONS LIS25270
51 28 C----- LIS25280
51 29          COMMON IP, JP, IBR, IPP, JPP, ATENDA, LIS25290
51 30          C NFREEP, NFREEB, LASTR, ARG, ARG2, ARG3, LIS25300
51 31          C ARG4, MODE, ALIST, SUBR0, SUBR1, SUBR2, LIS25310
51 32          C SUBR3, A2, A8, A9, LAMBDA, LABEL, LIS25320
51 33          C FUNARG, NIL, EXPR, FEXPR, SUBR, FSUBR, LIS25330
51 34          C T, SYS, RDPNT, PRTPNT, BSTACK(5), LIS25340
51 35          C DREG(24) LIS25350
51 36          COMMON /CONST/ NATCM,NFREET,NSTACK,NATOMP,NUMADD,UNDEF,LUNIN, LIS25360
51 37          C LUNUT,IBIG,IRANGE LIS25370
51 38          COMMON /SYMBCL/ SPACE,LPAR,RPAR,COMMA,DOT,LBRACK,RBRACK,IPROC, LIS25380
51 39          C IPLUS,IMINUS,IFIG(10),ATEND,ALLSP,CC LIS25390
51 40          COMMON /CARSTR/ NILCAR(2),CAR(10) LIS25400
51 41          COMMON /CDRSTR/ NILCDR(2),CDR(10) LIS25410
51 42          COMMON/ATCM1/ PNAME(4008) LIS25420
51 43          COMMON/ATCM2/ PNAME1,PNAME2 LIS25430
51 44          LOGICAL*1 PNAME LIS25440
51 45          COMMON /STKSTR/ STACK(10) LIS25450
51 46          INTEGER PNAME1, PNAME2, SPACE, RPAR, COMMA, DOT, LIS25460
51 47          I ALLSP, RBRACK, UNDEF, ATEND, ARG, ARG2, LIS25470
51 48          I ARG3, ARG4, ALIST, SUBR0, SUBR1, SUBR2, LIS25480
51 49          I SUBR3, A2, A8, A9, FUNARG, EXPR, LIS25490
51 50          I FEXPR, SUBR, FSUBR, T, SYS, RDPNT, LIS25500
52 1          I PRTFNT, ATENDA, BSTACK, DREG, CC LIS25510
52 2          INTEGER PNAMEP,FNAMEN LIS25520
52 3          INTEGER*2 CAR,CDR,STACK,NILCAR,NILCDR LIS25530
52 4 COMMON AND INTEGER DECLARATIONS ENDED LIS25540
52 5 C----- LIS25550
52 6          IND = 1 FOR POSITIVE NUMBER, -1 OTHERWISE LIS25560
52 7          LAB = 1 IF AN ATOM IS BEING PROCESSED, 2 IF FIRST CHARACTER LIS25570
52 8          READ WAS A BREAK CHARACTER LIS25580
52 9          NUM = 1 WHEN ATCM NOT AN INTEGER, 2 ATOM IS AN INTEGER, LIS25590
52 10         3 ATOM IS NOT OF KNOWN TYPE YET OR ELSE A + OR - LIS25600
52 11         LOGICAL*1 IB(80) LIS25610
52 12         IF(IATOM) 101,103,103 LIS25620
52 13         101 RETURN LIS25630
52 14         103 IND=1 LIS25640
52 15             K1=0 LIS25650
52 16             LAB=2 LIS25660
52 17             ISUM=0 LIS25670
52 18             IFLAG = 0 LIS25680
52 19             NUM=3 LIS25690
52 20             1 IF(CC-IPLUS)2,26,2 LIS25700
52 21             2 IF(CC-IMINUS)3,20,3 LIS25710
52 22             20 IND=-1 LIS25720
52 23             26 CALL SBYT(CC,IB,K1) LIS25730
52 24             GO TO 35 LIS25740
52 25             3 IF(CC-SPACE) 4,10,4 LIS25750
52 26             10 GO TO (40,11),LAB LIS25760
52 27             11 CALL SHIFT(CC) LIS25770
52 28             GO TO 1 LIS25780

```

52	29	4	IR=2	LIS25790
52	30		IF(CC-LPAR)5,25,5	LIS25800
52	31	5	IR=3	LIS25810
52	32		IF(CC-RPAR)6,25,6	LIS25820
52	33	6	IR=4	LIS25830
52	34		IF(CC-DOT)7,25,7	LIS25840
52	35	7	IR=5	LIS25850
52	36		IF(CC-LERACK) 8,25,8	LIS25860
52	37	8	IR=6	LIS25870
52	38		IF(CC-RBRACK) 9,25,9	LIS25880
52	39	9	IR=7	LIS25890
52	40		IF(CC-IFROC) 91,25,91	LIS25900
52	41	25	GO TO (40,50),LAB	LIS25910
52	42	91	CALL SBYT(CC,IB,K1)	LIS25920
52	43		GO TO (35,30,30), NUM	LIS25930
52	44	30	IF (CC-IFIG(1)) 36,32,31	LIS25940
52	45	31	IF (CC-IFIG(10)) 32,32,36	LIS25950
52	46	32	NUM =2	LIS25960
52	47		ISUM = ISUM*10+(CC-IFIG(1))/16777216	LIS25970
52	48		IF (ISUM-IRANGE) 35,35,321	LIS25980
52	49	321	ISUM= IRANGE	LIS25990
52	50		IFLAG = 1	LIS26000
53	1		GO TO 35	LIS26010
53	2	36	NUM=1	LIS26020
53	3	35	LAB=1	LIS26030
53	4		CALL SHIFT(CC)	LIS26040
53	5		GO TO 3	LIS26050
53	6	40	GC TO (41,42,41), NUM	LIS26060
53	7	41	IATOM=MATOM(IB,K1)	LIS26070
53	8		IRET=1	LIS26080
53	9		RETURN	LIS26090
53	10	42	ISUM = ISUM*IND	LIS26100
53	11		IF (IFLAG.GT.0) WRITE(3,100) ISUM	LIS26110
53	12	100	FORMAT (43H INTEGER HAS EXCEEDED LIMIT - VALUE SET TO ,16)	LIS26120
53	13		IATOM = ISUM+NUMADD	LIS26130
53	14		IRET=1	LIS26140
53	15		RETURN	LIS26150
53	16	50	IRET=IR	LIS26160
53	17		CALL SHIFT(CC)	LIS26170
53	18		RETURN	LIS26180
53	19		END	LIS26190
TAPE MARK				
54	1		DEFINE(((DEFIN(LAMBDA(X Y)(PROG(Z U)(SETQ U(INUNIT X))	DEFIN010
55	1		(SETQ Z Y) (CCND(Y(GO A))) B(COND((EVAL(READ)NIL)(GO B))) (GO C)	DEFIN020
56	1		A(DEFIN*(READ)Y)(COND((DEFIN* Y)(GO A))(T(GO C)))C(INUNIT U)(RETURN Z)))	DEFIN030
57	1		((DEFIN*(LAMBDA(X Y)(COND((NULL Y)NIL)((EQ(CAR Y)(CAAR(CADR X)))	DEFIN040
58	1		(PROGN(EVAL X NIL)(RPLACA Y NIL)))(T(DEFIN* X(CDR Y))))))	DEFIN050
59	1		(DEFIN*(LAMBDA(Z)(COND((NULL Z)NIL)((NULL(CAR Z))(DEFIN*(CDR Z)))	DEFIN060
60	1		(T T))))))	DEFIN070
61	1		DEFINE ((
62	1		(MAPCAR (LAMBDA (F LIST)	
63	1		(CCND ((NULL LIST) NIL)	
64	1		(T (CONS (F (CAR LIST))	
65	1		(MAPCAR F(CDR LIST))	
66	1)))	
67	1		DEFINE ((
68	1		(LENGTH(LAMBDA(M) (COND ((NULL M)0)	
69	1		(T(ADD1(LENGTH(CDR M)))))))	
70	1		DEFINE ((
71	1		(SEARCH (LAMBDA (Z P F U) (CCND ((NULL Z) (U Z))	

```

72 1 ((P Z) (F Z))
73 1 (T (SEARCH (CDR Z) P F U)) ) )
74 1 (SUBLIS (LAMBDA (X Y) (COND ((NULL X) Y) ((NULL Y) Y)
75 1 ( T (SEARCH X
76 1 (QUOTE (LAMBDA (J) (EQUAL Y (CAAR J)) ) )
77 1 (QUOTE (LAMBDA (J) (CDAR J) ) )
78 1 (QUOTE (LAMBDA (J) (COND ((ATOM Y) Y)
79 1 (T (CONS (SUBLIS X (CAR Y)) (SUBLIS X (CDR Y))))
80 1 ) ) )
81 1 ) ) )
82 1 ))
83 1 ))
84 1 DEFINE(((DELETE(LAMBDA(X)(PROGN(RPLACD X NIL)(COND((ATOM X)NIL)
85 1 (T(RPLACA X NIL)))))) )) )
86 1 DEFINE((
87 1 (UNION (LAMBDA (X Y) (COND ((NULL X) Y) ((MEMBER (CAR X)Y)
88 1 (UNION (CDR X)Y)) (T (CONS (CAR X) (UNION (CDR X) Y)))) )) )
89 1 (INTERSECTION (LAMBDA (X Y) (COND ((NULL X) X) ((MEMBER (CAR X) Y)
90 1 (CONS (CAR X) (INTERSECTION (CDR X) Y)))) (T (INTERSECTION (CDR X)Y))
91 1 )))
92 1 ))
93 1 DEFINE(((ATTRIB(LAMBDA(X Y) (PROGN (NCONC X Y) Y) )) ))
94 1 DEFINE(((SUBST (LAMBDA (X Y Z) (COND ((EQUAL Y Z) X)
95 1 ((ATCM Z) Z)
96 1 (T (CONS (SUBST X Y (CAR Z)) (SUBST X Y (CDR Z)))) ) ) ) )
97 1 DEFINE(((CLIST (LAMBDA (X) (COND ((NULL X)X) ((ATOM X) (LIST X))
98 1 (T X) ) ) ) )
99 1 DEFLIST(((QLIST(LAMBDA(X Y)(COPY X) )))FEXPR)
100 1 DEFINE(((COPY(LAMBDA(X) (COND ((NULL X)X) ((ATOM X)X)
101 1 (T (CONS (COPY (CAR X)) (COPY (CDR X)))) ) ) ) ) )
102 1 DEFLIST((
103 1 (TRACE (LAMBDA (X Y)(*TRC X)))
104 1 (UNTRACE (LAMBDA (X Y)(*UTRC X))) )FEXPR)
105 1 DEFINE((
106 1 (*TRC (LAMBDA (X) (COND
107 1 ((NULL X) X)(T(CONS(COND((*CHK(CDAR X))(CAR X))(CAR X))
108 1 (T NIL))(*TRC (CDR X))))))
109 1 (*UTRC (LAMBDA (X)(COND
110 1 ((NULL X) X)(T(CONS(COND((*UNK(CDAR X))(CAR X))(T NIL))
111 1 (*UTRC (CDR X))))))
112 1 (*CPL(LAMBDA (X Y)(RPLACA X(LIST(QUOTE LAMBDA)(CAR(CDAR X))
113 1 (LIST (QUOTE PROGN)
114 1 (LIST(QUOTE *TRE)(LIST(QUOTE QUOTE)Y)
115 1 (CONS(QUOTE LIST)(CAR(CDAR X))))(LIST
116 1 (QUOTE *TRR)(LIST(QUOTE QUOTE)Y)(CONS(CAR X)(CAR(CDAR X))))))))))
117 1 (*CHK(LAMBDA(X Y)(COND((NULL X)NIL)
118 1 ((EQ (CAR X)(QUOTE EXPR))(*CPL(CDR X)Y))
119 1 ((EQ (CAR X)(QUOTE FEXPR))(*CPL(CDR X)Y))
120 1 (T(*CHK(CDDR X)Y))))))
121 1 (*TRE(LAMBDA (X Y)(PRINT (LIST X (QUOTE ARG )Y))))
122 1 (*TRR (LAMBDA (X Y)(CAR(CDDR(PRINT(LIST X (QUOTE VALUE)Y))))))
123 1 (*UNK(LAMBDA (X)(COND((NULL X)NIL)
124 1 ((EQ (CAR X) (QUOTE EXPR))(*UPL X))
125 1 ((EQ (CAR X) (QUOTE FEXPR))(*UPL X))(T(*UNK(CDDR X))))))
126 1 (*UPL(LAMBDA(X)(RPLACD X(LIST(CAAR(CDDR(CADR(CDAR(CDDR(CADR X))))))))))
127 1 ))

```

TAPE MARK
TAPE MARK

