File created: 15-Jun-90 13:37:15 **{DSK}<usr>local>lde>lispcore>internal>library>COMPTEST.;2**

changes to: (VARS COMPTESTCOMS)

previous date: 17-Jun-88 18:52:58 {DSK}<usr>local>lde>lispcore>internal>library>COMPTEST.;1

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

;;


```
(RPAQQ COMPTESTCOMS
       [(FNS MAKEFORM TESTER TEST1 TRY FAILTEST TESTVALS TESTIFSAME TESTPAIRS EVALVARS TRYTEST)
        (ADDVARS (TESTS TESTVAR TESTTYPE TESTRET TESTOP TESTNUM TESTMORE TESTMISC TESTMAP TESTLINK TESTJUMP
                        TESTFNX TESTFN TESTEDIT TESTCONS TESTCAR TESTCALL TESTC2 TESTC TESTBIND TESTAT TEST3))
        (COMS (FNS IVAR PVAR VARSWAP VARSWAP2 VARSWAP3 VARSWAP4 IVAR3 IVARX FVAR)
              (VARS TESTVAR)
              (FNS .GETPROPLST)
              (FNS .LISTP .NLISTP .LITATOM .FLOATP .FIXP .NUMBERP .SMALLP .STACKP .ARRAYP .NLITATOM .NFLOATP
                   .NNUMBERP .NFIXP .NSMALLP .NARRAYP .NSTACKP .NZEROP .ZEROP .STRINGP .NSTRINGP .IGREATERP
                   .NIGREATERP .ILESSP .NILESSP .ATOM .NATOM .EQ .NEQ .NULL .NNULL .IEQP .NIEQP .ORLISTP
                   .ANDLISTP .ORATOM .ANDATOM .ORZEROP .ORNULL .ORARRAYP .ANDARRAYP .ANDNLISTP .ANDNATOM
                   .ORFLOATP .ANDFLOATP)
              (VARS TESTTYPE)
              (FNS .CONDRET .CONDRET2 TESTRESUME GETLEAVES)
              (VARS TESTRET)
              (FNS .TESTARG .SET .EVALV1 .EVALV2 .SUM .FIX .LIST .NLSETQ .EQUAL .SETX)
              (VARS TESTOP)
              (FNS .ITIMES .IPLUS .IQUOTIENT .IREMAINDER .ADD1 .SUB1 .LLSH .LRSH .LSH .RSH .LOGAND .LOGOR
                   .LOGXOR .IDIFFERENCE .NT1 .NT2 .NT3)
              (VARS TESTNUM)
              (FNS .NCONC ..NCONC .AND .FRPLNODE .OR .FRPLNODE2 ..NCONC3 .NCONC3 SELECTTEST .MKLIST .EQMEMB
                   .NCONC1 .GETPROPLIST .SETPROPLIST .FGETD ..FRPLNODE2 !AND !OR)
              (VARS TESTMORE)
              (FNS !ADD1VAR !APPEND APPEND2 !ASSOC !ATTACH !CHANGEPROP !COPY !DEFLIST !DREMOVE !DREVERSE DREV
                   !DSUBST !EVERY !GETP !INTERSECTION !LAST !LASTN !LDIFF !LENGTH !LISTGET !LSUBST !MAP !GET
                   !GETLIS !MEMB !NTH .COLLCT .ENDCOLLCT MYAPPEND1 MYAPPEND2 COLLCT ENDCOLLCT .ATTACH .APPEND0
                   .APPEND1 .APPEND2 .APPEND3 .APPEND4 .MAPCGETP)
              (VARS TESTMISC)
              (FNS .MAP .MAPC .MAPCEFF)
              (VARS TESTMAP)
              (FNS .FPLUS .FTIMES .FDIFFERENCE .FQUOTIENT)
              (VARS TESTLINK)
              (FNS NONLOCALGO CNTDWN JUMPAROUND)
              (VARS TESTJUMP)
              (FNS FN2 FN3 FN1 .IVAR)
              (VARS TESTFNX)
              (FNS .GETP .GETPROP .RPLACA .RPLACD .FRPLACA .GET .FRPLACD .ASSOC .LENGTH .LAST .GETHASH .FMEMB)
              (VARS TESTFN)
              (VARS TESTEDIT)
              (FNS .CONS LIST0 LIST1 LIST2 LIST3 LIST4 LIST5 LIST6 LIST7 LIST8 LIST9 LIST10 LIST11 LIST12 LIST13
                   )
              (VARS TESTCONS)
              (FNS .CAR .CDR .CAAR .CDAR .CADR .CDDR .CAAAAR)
              (VARS TESTCAR)
              (FNS .LAM0 ..LAM0 .LAM1 ..LAM1 .NLAML LAM1LOC)
              (VARS TESTCALL)
              (FNS .PROGS .SPEC .COND .DELBIND)
              (VARS TESTC2)
              (FNS T1 T0 TT TNIL T-1 T2 T12 T377 T400Q T-400 TSTR .NILARGS)
              (VARS TESTC)
              (FNS .BIND0 .BIND1 .BIND2 .BIND3 .BIND4 .BINDASSOC .BIND5 .BINDPOP)
              (VARS TESTBIND)
              (VARS TESTAT)
              (FNS LAM0 LAM1 LAMA NLAML NLAMA)
              (FNS .SELECTQ .SUBFNS .MISC .FORTEST .BIGCOND .RECORDTEST .PROGRETURN .ALWAYSFALSE .ALWAYSTRUE
                   .EQ1 .EQ2 .EQ3)
              (VARS TEST3))
        (DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVARS (ADDVARS (NLAMA NLAMA TESTIFSAME TESTVALS)
                                                                               (NLAML NLAML !ADD1VAR .NLSETQ
                                                                                      TESTPAIRS)
                                                                               (LAMA LAMA !APPEND .LIST .SUM
                                                                                     .TESTARG])

(DEFINEQ

(MAKEFORM
  [LAMBDA (FORM)                                          ; Edited  8-Apr-88 12:52 by amd
    (SUBST FORM 'FORM '(LIST 1 FORM (COND
                                    (FORM 2)
                                    ((NOT FORM)
                                     3)
```

```
                                    (T 4))
                            (OR FORM 5)
                            (AND FORM 6)
                            (PROGN FORM 7])
```

(TESTER

```
  [LAMBDA (TESTLST)                                          ; Edited 8-Apr-88 16:20 by amd
    (HANDLER-BIND [(CL:ERROR #'(LAMBDA (C)
                                 (LET [(RESTART (CONDITIONS:FIND-RESTART 'COMPTEST-FAIL]
                                      (IF (NULL RESTART)
                                          THEN (HELP "Error signalled while not below TRY")
                                        ELSE (CONDITIONS:INVOKE-RESTART RESTART C]
          (TEST1 TESTLST])
```

(TEST1

```
  [LAMBDA (TESTLST)                                          ; Edited 8-Apr-88 12:53 by amd
    (COND
       ((NULL TESTLST)
        (MAPC TESTS 'TEST1))
       ((LITATOM TESTLST)
        (PRINT TESTLST T)
        (TEST1 (EVALV TESTLST)))
       (T (MAPC TESTLST 'EVAL])
```

(TRY

```
  [LAMBDA (FORM)                                             ; Edited 8-Apr-88 16:19 by amd
    (CONDITIONS:RESTART-CASE (EVAL FORM)
           (COMPTEST-FAIL (CONDITION)
                  (LIST :ERROR (CL:PRINC-TO-STRING CONDITION])
```

(FAILTEST

```
  [LAMBDA NIL
    (PRIN2 TEST T T)
    (PRIN1 " failed.
           " T])
```

(TESTVALS

```
  [NLAMBDA L                                                 ; Edited 17-Jun-88 18:50 by amd
    (PROG ((A 'A.TOP)
           (B 'B.TOP)
           (C 'C.TOP)
           (D 'D.TOP)
           (E 'E.TOP)
           (F 'F.TOP)
           (G 'G.TOP)
           (H 'H.TOP)
           (I 'I.TOP))
          (MAP L [FUNCTION (LAMBDA (X)
                             (OR (AND [CL:EQUALP [SETQ V1 (TRY (SETQ TEST (CAR X]
                                                       (SETQ V2 (TRY (CADR X]
                                      (EQ (TYPENAME V1)
                                          (TYPENAME V2)))
                                 (FAILTEST]
               (FUNCTION CDDR])
```

(TESTIFSAME

```
  [NLAMBDA FNVALS                                            ; Edited 8-Apr-88 12:53 by amd
    (MAPC FNVALS (FUNCTION (LAMBDA (L V1 V2)
                             (OR (AND [EQUAL [SETQ V1 (TRY (SETQ TEST (CONS (CAR L)
                                                                            (CDDR L]
                                                       (SETQ V2 (TRY (CDR L]
                                      (EQ (TYPENAME V1)
                                          (TYPENAME V2)))
                                 (FAILTEST])
```

(TESTPAIRS

```
  [NLAMBDA (FNL VALLST)                                      ; Edited 8-Apr-88 12:53 by amd
    (MAPC FNL (FUNCTION (LAMBDA (FNPR)
                          (MAP VALLST (FUNCTION (LAMBDA (VLST)
                                                  (MAPC VLST
                                                        (FUNCTION (LAMBDA (VLST2)
                                                                    (OR
                                                                     (AND [EQUAL [SETQ V1
                                                                                  (TRY (SETQ TEST
                                                                                         (LIST (CAR FNPR)
                                                                                               (CAR VLST)
                                                                                               VLST2]
                                                                                (SETQ V2
                                                                                 (TRY (CONS (CDR FNPR)
                                                                                            (CDR TEST]
```

```
                                                        (EQ (TYPENAME V1)
                                                            (TYPENAME V2)))
                                                    (FAILTEST])
```

(**EVALVARS**

```
  [LAMBDA NIL                                           ; Edited 8-Apr-88 12:54 by amd
                                                        ; lmm: 22-JUN-76 0 56
    (MAPCAR '(A B C D E F G H)
           (FUNCTION EVALV])
```

(**TRYTEST**

```
  [LAMBDA (FORM1 FORM2)                                 ; Edited 8-Apr-88 12:54 by amd
                                                        ; lmm: 24-JUN-76 4 41
    (OR (EQUAL (TRY (SETQ TEST FORM1))
               (TRY FORM2))
        (FAILTEST])
```

)

```
(ADDTOVAR TESTS TESTVAR TESTTYPE TESTRET TESTOP TESTNUM TESTMORE TESTMISC TESTMAP TESTLINK TESTJUMP TESTFNX
                      TESTFN TESTEDIT TESTCONS TESTCAR TESTCALL TESTC2 TESTC TESTBIND TESTAT TEST3)

(DEFINEQ
```

(**IVAR**

```
  [LAMBDA (A B C D E F G H I J K L M N O)
    (DECLARE (SPECVARS))
    (DECLARE (LOCALVARS . T))
    (LIST A B C D E F G H I J K L M N O])
```

(**PVAR**

```
  [LAMBDA (A B C D E F G H I J K L M N O)                ; Edited 8-Apr-88 12:54 by amd
                                                        ; lmm: 19-JUN-76 0 14
    (PROG (X Y Z)
          (DECLARE (LOCALVARS . T))
          (RETURN (LIST A B C D E F G H I J K L M N O])
```

(**VARSWAP**

```
  [LAMBDA (X Y Z W)                                     ; Edited 8-Apr-88 12:54 by amd
                                                        ; lmm: 19-JUN-76 1 18
    (DECLARE (LOCALVARS . T))
    (SETQ X Y)
    (SETQ Y Z)
    (SETQ Z W)
    (SETQ W 0)
    (LIST X Y Z W])
```

(**VARSWAP2**

```
  [LAMBDA (X Y Z W)                                     ; Edited 8-Apr-88 12:55 by amd
                                                        ; lmm: 19-JUN-76 1 31
    (PROG NIL
          (DECLARE (LOCALVARS . T))
          (SETQ X Y)
          (SETQ Y Z)
          (SETQ Z W)
          (SETQ W 0)
          (RETURN (LIST X Y Z W])
```

(**VARSWAP3**

```
  [LAMBDA NIL                                           ; Edited 8-Apr-88 12:55 by amd
                                                        ; lmm: 19-JUN-76 1 34
    (PROG NIL
          (SETQ X Y)
          (SETQ Y Z)
          (SETQ Z W)
          (SETQ W 0)
          (RETURN (LIST X Y Z W])
```

(**VARSWAP4**

```
  [LAMBDA NIL                                           ; Edited 8-Apr-88 12:55 by amd
                                                        ; lmm: 19-JUN-76 1 45
    (SETQ X Y)
    (SETQ Y Z)
    (SETQ Z W)
    (SETQ W 0)
    (LIST X Y Z W])
```

(**IVAR3**

```
  [LAMBDA (A B C D E F G H I J K L M N O)                ; Edited 8-Apr-88 12:55 by amd
```

```
                                                                    ; lmm: 19-JUN-76 3 24
    (DECLARE (SPECVARS))
    (DECLARE (LOCALVARS . T))
    (PROG (Z W)
          (RETURN (PROG (X Y)
                        (RETURN (LIST A B C D E F G H I J K L M N O])
```

(IVARX

```
  [LAMBDA (A B C)                                                   ; Edited  8-Apr-88 12:55 by amd
                                                                    ; lmm: 24-JUN-76 9 1
    (DECLARE (LOCALVARS . T))
    (PROG ((D (CONS 1 A))
           (E (CONS 2 B))
           (F (CONS 3 C)))
          (RETURN (PROG ((H (CONS 4 D))
                         (I (CONS 5 E))
                         (J (CONS 6 F)))
                        (RETURN (LIST A B C D E F H I J])
```

(FVAR

```
  [LAMBDA NIL                                                       ; Edited  8-Apr-88 12:55 by amd
                                                                    ; lmm: 24-JUN-76 13 28
    (CONS F1 F2])

)

(RPAQQ TESTVAR
       [(TESTVALS (IVAR 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15)
              '(1 2 3 4 5 6 7 8 9 10 11 12 13 14 15)
              (PVAR 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15)
              '(1 2 3 4 5 6 7 8 9 10 11 12 13 14 15)
              (IVAR3 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15)
              '(1 2 3 4 5 6 7 8 9 10 11 12 13 14 15)
              (VARSWAP 1 2 3 4)
              '(2 3 4 0)
              (VARSWAP2 1 2 3 4)
              '(2 3 4 0)
              (PROG ((X 1)
                     (Y 2)
                     (Z 3)
                     (W 4))
                    (RETURN (VARSWAP3)))
              '(2 3 4 0)
              (PROG ((X 1)
                     (Y 2)
                     (Z 3)
                     (W 4))
                    (RETURN (VARSWAP4)))
              '(2 3 4 0)
              (IVARX -3 -2 -1)
              '(-3 -2 -1 (1 . -3)
                   (2 . -2)
                   (3 . -1)
                   (4 1 . -3)
                   (5 2 . -2)
                   (6 3 . -1])

(DEFINEQ

(.GETPROPLST
  [LAMBDA (X Y Z)
    (LIST 1 (GETPROPLIST X)
          (PROG1 3 (GETPROPLIST X])

)

(DEFINEQ

(.LISTP
  [LAMBDA (X)
    (LISTP X])

(.NLISTP
  [LAMBDA (X)
    (NLISTP X])

(.LITATOM
  [LAMBDA (X)                                                       ; Edited  8-Apr-88 12:56 by amd
                                                                    ; lmm: 18-JUN-76 14 59
    (LITATOM X])

(.FLOATP
```

```
  [LAMBDA (X)                                          ; Edited  8-Apr-88 12:56 by amd
                                                       ; lmm: 18-JUN-76 14 59
    (FLOATP X])
```

(**.FIXP**
```
  [LAMBDA (X)                                          ; Edited  8-Apr-88 12:56 by amd
                                                       ; lmm: 18-JUN-76 14 59
    (FIXP X])
```

(**.NUMBERP**
```
  [LAMBDA (X)                                          ; Edited  8-Apr-88 12:56 by amd
                                                       ; lmm: 18-JUN-76 15 1
    (NUMBERP X])
```

(**.SMALLP**
```
  [LAMBDA (X)                                          ; Edited  8-Apr-88 12:56 by amd
                                                       ; lmm: 18-JUN-76 15 7
    (SMALLP X])
```

(**.STACKP**
```
  [LAMBDA (X)                                          ; Edited  8-Apr-88 12:57 by amd
                                                       ; lmm: 18-JUN-76 15 11
    (STACKP X])
```

(**.ARRAYP**
```
  [LAMBDA (X)                                          ; Edited  8-Apr-88 12:57 by amd
                                                       ; lmm: 18-JUN-76 15 11
    (ARRAYP X])
```

(**.NLITATOM**
```
  [LAMBDA (X)                                          ; Edited  8-Apr-88 12:57 by amd
                                                       ; lmm: 18-JUN-76 15 25
    (NOT (LITATOM X])
```

(**.NFLOATP**
```
  [LAMBDA (X)
    (NOT (FLOATP X])
```

(**.NNUMBERP**
```
  [LAMBDA (X)
    (NOT (NUMBERP X])
```

(**.NFIXP**
```
  [LAMBDA (X)
    (NOT (FIXP X])
```

(**.NSMALLP**
```
  [LAMBDA (X)
    (NOT (SMALLP X])
```

(**.NARRAYP**
```
  [LAMBDA (X)
    (NOT (ARRAYP X])
```

(**.NSTACKP**
```
  [LAMBDA (X)
    (NOT (STACKP X])
```

(**.NZEROP**
```
  [LAMBDA (X)
    (NOT (ZEROP X])
```

(**.ZEROP**
```
  [LAMBDA (X)                                          ; Edited  8-Apr-88 12:58 by amd
                                                       ; lmm: 18-JUN-76 15 28
    (ZEROP X])
```

(**.STRINGP**
```
  [LAMBDA (X)
    (STRINGP X])
```

(.**NSTRINGP**
```
  [LAMBDA (X)
    (NOT (STRINGP X])
```

(.**IGREATERP**
```
  [LAMBDA (X Y)
    (IGREATERP X Y])
```

(.**NIGREATERP**
```
  [LAMBDA (X Y)
    (NOT (IGREATERP X Y])
```

(.**ILESSP**
```
  [LAMBDA (X Y)
    (ILESSP X Y])
```

(.**NILESSP**
```
  [LAMBDA (X Y)
    (NOT (ILESSP X Y])
```

(.**ATOM**
```
  [LAMBDA (X)
    (ATOM X])
```

(.**NATOM**
```
  [LAMBDA (X)
    (NOT (ATOM X])
```

(.**EQ**
```
  [LAMBDA (X Y)
    (EQ X Y])
```

(.**NEQ**
```
  [LAMBDA (X Y)
    (NEQ X Y])
```

(.**NULL**
```
  [LAMBDA (X)
    (NULL X])
```

(.**NNULL**
```
  [LAMBDA (X)
    (COND
       ((NULL X)
        NIL)
       (T T])
```

(.**IEQP**
```
  [LAMBDA (X Y)
    (IEQP X Y])
```

(.**NIEQP**
```
  [LAMBDA (X Y)                          ; Edited  8-Apr-88 12:59 by amd
                                         ; lmm: 19-JUN-76 3 14

    (NOT (IEQP X Y])
```

(.**ORLISTP**
```
  [LAMBDA (Y X)                          ; Edited  8-Apr-88 12:59 by amd
                                         ; lmm: 24-JUN-76 4 45

    (OR (LISTP X)
        Y])
```

(.**ANDLISTP**
```
  [LAMBDA (Y X)                          ; Edited  8-Apr-88 12:59 by amd
                                         ; lmm: 24-JUN-76 4 45

    (AND (LISTP X)
         Y])
```

(.**ORATOM**
```
  [LAMBDA (Y X)                          ; Edited  8-Apr-88 13:00 by amd
                                         ; lmm: 30-JUN-76 13 27

    (OR (ATOM X)
```

```
        Y])
```

(**.ANDATOM**

```
  [LAMBDA (Y X)                                          ; Edited  8-Apr-88 13:00 by amd
                                                         ; lmm: 30-JUN-76 13 27
    (AND (ATOM X)
         Y])
```

(**.ORZEROP**

```
  [LAMBDA (Y X)                                          ; Edited  8-Apr-88 13:00 by amd
                                                         ; lmm: 24-JUN-76 4 47
    (OR (EQ X 0)
        Y])
```

(**.ORNULL**

```
  [LAMBDA (Y X)                                          ; Edited  8-Apr-88 13:00 by amd
                                                         ; lmm: 24-JUN-76 4 47
    (OR (NULL X)
        Y])
```

(**.ORARRAYP**

```
  [LAMBDA (Y X)                                          ; Edited  8-Apr-88 13:00 by amd
                                                         ; lmm: 24-JUN-76 4 47
    (OR (ARRAYP X)
        Y])
```

(**.ANDARRAYP**

```
  [LAMBDA (Y X)                                          ; Edited  8-Apr-88 13:00 by amd
                                                         ; lmm: 24-JUN-76 13 9
    (AND (ARRAYP X)
         Y])
```

(**.ANDNLISTP**

```
  [LAMBDA (Y X)                                          ; Edited  8-Apr-88 13:00 by amd
                                                         ; lmm: 30-JUN-76 13 26
    (AND (NOT (LISTP X))
         Y])
```

(**.ANDNATOM**

```
  [LAMBDA (Y X)                                          ; Edited  8-Apr-88 13:00 by amd
                                                         ; lmm: 30-JUN-76 13 28
    (AND (NOT (ATOM X))
         Y])
```

(**.ORFLOATP**

```
  [LAMBDA (Y X)                                          ; Edited  8-Apr-88 13:01 by amd
                                                         ; lmm: 30-JUN-76 13 28
    (OR (FLOATP X)
        Y])
```

(**.ANDFLOATP**

```
  [LAMBDA (Y X)                                          ; Edited  8-Apr-88 13:01 by amd
                                                         ; lmm: 30-JUN-76 13 28
    (AND (FLOATP X)
         Y])
)

(RPAQQ TESTTYPE
       [[MAPC '((.NLISTP .LISTP LISTP .ORLISTP .ANDLISTP)
                (.NATOM .ATOM ATOM .ORATOM .ANDATOM .ANDNATOM)
                (.NLITATOM .LITATOM LITATOM)
                (.NFLOATP .FLOATP FLOATP .ORFLOATP .ANDFLOATP)
                (.NNUMBERP .NUMBERP NUMBERP)
                (.NFIXP .FIXP FIXP)
                (.NSMALLP .SMALLP SMALLP)
                (.NARRAYP .ARRAYP ARRAYP .ORARRAYP .ANDARRAYP)
                (.NSTACKP .STACKP STACKP)
                (.NZEROP .ZEROP ZEROP .ORZEROP)
                (.NSTRINGP .STRINGP STRINGP)
                (.NNULL .NULL NULL .ORNULL))
              (FUNCTION (LAMBDA (L)
                                (MAPC [CONS (LIST)
                                            (CONS (LIST (ARRAY 2))
                                                  (CONS (LIST (STKNTH 0 T))
                                                        '((-1)
                                                          (0)
                                                          (1)
```

```
                                                    (1.0)
                                                    (100000)
                                                    ('A)
                                                    ('(A))
                                                    ("foo")
                                                    (NIL)
                                                    (T]
                                          (FUNCTION (LAMBDA (ARGL TX)
                                                            (TRYTEST (CONS (CADR L)
                                                                           ARGL)
                                                                     (SETQ TX (CONS (CADDR L)
                                                                                    ARGL)))
                                                            (TRYTEST (CONS (CAR L)
                                                                           ARGL)
                                                                     (LIST 'NOT TX))
                                                            (AND (CADDDR L)
                                                                 (TRYTEST (CONS (CADDDR L)
                                                                                (CONS 74 ARGL))
                                                                          (LIST 'OR TX 74)))
                                                            (AND (CAR (CDDDDR L))
                                                                 (TRYTEST (CONS (CAR (CDDDDR L))
                                                                                (CONS 74 ARGL))
                                                                          (LIST 'AND TX 74)))
                                                            (AND (CADR (CDDDDR L))
                                                                 (TRYTEST (CONS (CADR (CDDDDR L))
                                                                                (CONS 74 ARGL))
                                                                          (LIST 'AND (LIST 'NOT TX)
                                                                                74]
        (MAPC '((.NIGREATERP .IGREATERP IGREATERP)
                (.NILESSP .ILESSP ILESSP)
                (.NIEQP .IEQP IEQP)
                (.NEQ .EQ EQ))
              (FUNCTION (LAMBDA (L)
                                (MAPC '((1 -1)
                                        (-1 1)
                                        (100000 1)
                                        (100.0 1)
                                        (1 100.0)
                                        (100.0 200.0)
                                        (1 100000)
                                        (300000 -300000)
                                        (-300000 300000)
                                        (100000 100000))
                                      (FUNCTION (LAMBDA (ARGS)
                                                        (TRYTEST (CONS (CADR L)
                                                                       ARGS)
                                                                 (CONS (CADDR L)
                                                                       ARGS))
                                                        (TRYTEST (CONS (CAR L)
                                                                       ARGS)
                                                                 (LIST 'NOT (CONS (CADDR L)
                                                                                  ARGS])

(DEFINEQ

(.CONDRET
  [LAMBDA (X)                                                    ; Edited  8-Apr-88 13:01 by amd
    (PROG NIL
          (COND
             (X (RETURN (CONS)))
             (T (RETURN 3])

(.CONDRET2
  [LAMBDA NIL                                                    ; Edited  8-Apr-88 13:01 by amd
    (PROG (X)
          (COND
             ((RETURN X)
              T)
             (T 3])

(TESTRESUME
  [LAMBDA (STRUCTURE)
    (PROG (LEAF RESULT TESTPTR GETPTR)
          (COROUTINE TESTPTR GETPTR (GETLEAVES STRUCTURE GETPTR TESTPTR))
      LP  (COND
             ((SETQ LEAF (RESUME TESTPTR GETPTR))
              (SETQ RESULT (NCONC1 RESULT LEAF))
              (GO LP))
             (T (RETURN RESULT])

(GETLEAVES
  [LAMBDA (STRUC GETPTR TESTPTR)
    (COND
       ((LISTP STRUC)
```

```
        (GETLEAVES (CAR STRUC)
                   GETPTR TESTPTR)
        (GETLEAVES (CDR STRUC)
                   GETPTR TESTPTR))
       (STRUC (RESUME GETPTR TESTPTR STRUC])

)

(RPAQQ TESTRET
       [(TESTVALS (.CONDRET T)
                  '(NIL)
                  (.CONDRET)
                  3
                  (.CONDRET2 17)
                  NIL
                  [TESTRESUME '(1 (2 . 3)
                                  (4 . 5) . 6]
                  '(1 2 3 4 5 6])

(DEFINEQ

(.TESTARG
  [LAMBDA N                                          ; Edited 8-Apr-88 13:01 by amd
                                                     ; lmm: 22-JUN-76 22 59
    (LIST N (ARG N 1)
          (ARG N 2])

(.SET
  [LAMBDA (VAR VAL)                                  ; Edited 8-Apr-88 13:02 by amd
    (LIST 1 (SET VAR VAL)
          3])

(.EVALV1
  [LAMBDA (X)                                        ; Edited 8-Apr-88 13:02 by amd
                                                     ; lmm: 22-JUN-76 23 4
    (EVALV X])

(.EVALV2
  [LAMBDA (X Y)                                      ; Edited 8-Apr-88 13:02 by amd
                                                     ; lmm: 22-JUN-76 23 5
    (EVALV X Y])

(.SUM
  [LAMBDA N                                          ; Edited 8-Apr-88 13:02 by amd
                                                     ; lmm: 24-JUN-76 14 59
    (PROG ((I N)
           (V 0))
      LP   (COND
              ((ZEROP I)
               (RETURN V))
              (T (SETQ V (IPLUS V (ARG N I)))
                 (SETQ I (SUB1 I))
                 (GO LP])

(.FIX
  [LAMBDA (X)                                        ; Edited 8-Apr-88 13:02 by amd
                                                     ; lmm: 25-JUN-76 6 20
    (FIX X])

(.LIST
  [LAMBDA L                                          ; Edited 8-Apr-88 13:02 by amd
                                                     ; lmm: 30-JUN-76 14 50
    (AND (NOT (ZEROP L))
         (PROG ((I L)
                V)
           LP   (COND
                   ((EQ I 1)
                    (RETURN (CONS (ARG L 1)
                                  V)))
                   (T (SETQ V (CONS (ARG L I)
                                    V))
                      (SETQ I (SUB1 I))
                      (GO LP])

(.NLSETQ
  [NLAMBDA (FORM)                                    ; Edited 8-Apr-88 13:03 by amd
                                                     ; lmm: 1-JUL-76 10 44
    (NLSETQ (EVAL FORM])
```

```
(.EQUAL
  [LAMBDA (X Y)                                          ; Edited  8-Apr-88 13:03 by amd
    (LIST 1 (EQUAL X Y)
          3])


(.SETX
  [LAMBDA (X Y Z)
    (DECLARE (LOCALVARS . T))                            ; Edited  8-Apr-88 13:03 by amd
    (PROG (K L M)
          (COND
             (X (SETQ K -1))
             (T (SETQ K 3)))
          (COND
             (Y (SETQ L -4))
             (T (SETQ L 4)))
          (COND
             (Z (SETQ M -2))
             (T (SETQ M 2)))
          (RETURN (LIST K L M])

)

(RPAQQ TESTOP
       [(SETTOPVAL 'A 'A.TOPLEVEL)
        (TESTVALS (.TESTARG 10 4 12 'A 'B)
               '(5 10 4)
               (PROG (FOO)
                     (RETURN (LIST (.SET 'FOO 300)
                                   FOO)))
               '((1 300 3)
                 300)
               (.EVALV1 'A)
               'A.TOP
               (.EVALV2 'A)
               'A.TOP
               (.EVALV2 'A T)
               'A.TOPLEVEL
               (.SUM 34 -34)
               0
               (.SUM 10 9 8 -8 -9 -10)
               0
               (.FIX (SETQ TESTATOM 100000))
               TESTATOM
               (.FIX 1.3)
               1
               (.NLSETQ (ERROR!))
               NIL
               (.NLSETQ 'FOO)
               '(FOO)
               (.LIST)
               NIL
               (.LIST 1)
               '(1)
               (.LIST 1 2 3 4 5)
               '(1 2 3 4 5)
               (.EQUAL '((A)
                         "B" 4.0 "CDEFG")
                       '((A)
                         "B" 4 "CDEFG"))
               '(1 T 3)
               [.EQUAL '(("ABCDEFG" "IJ"))
                       '(("ABCDEGF" "IJ"]
               '(1 NIL 3)
               (.SETX T T T)
               '(-1 -4 -2)
               (.SETX)
               '(3 4 2])

(DEFINEQ

(.ITIMES
  [LAMBDA (X Y)
    (ITIMES X Y])


(.IPLUS
  [LAMBDA (X Y)
    (IPLUS X Y])


(.IQUOTIENT
  [LAMBDA (X Y)
    (IQUOTIENT X Y])


(.IREMAINDER
```

```
  [LAMBDA (X Y)
    (IREMAINDER X Y])
```

(**.ADD1**
```
  [LAMBDA (X)
    (ADD1 X])
```

(**.SUB1**
```
  [LAMBDA (X)
    (SUB1 X])
```

(**.LLSH**
```
  [LAMBDA (X Y)
    (LLSH X Y])
```

(**.LRSH**
```
  [LAMBDA (X Y)
    (LRSH X Y])
```

(**.LSH**
```
  [LAMBDA (X Y)
    (LSH X Y])
```

(**.RSH**
```
  [LAMBDA (X Y)
    (RSH X Y])
```

(**.LOGAND**
```
  [LAMBDA (X Y)
    (LOGAND X Y])
```

(**.LOGOR**
```
  [LAMBDA (X Y)
    (LOGOR X Y])
```

(**.LOGXOR**
```
  [LAMBDA (X Y)
    (LOGXOR X Y])
```

(**.IDIFFERENCE**
```
  [LAMBDA (X Y)
    (IDIFFERENCE X Y])
```

(**.NT1**
```
  [LAMBDA (X Y Z)
    (FTIMES (DIFFERENCE X Y)
           Z])
```

(**.NT2**
```
  [LAMBDA (X Y Z)
    (FTIMES (QUOTIENT X Y)
           Z])
```

(**.NT3**
```
  [LAMBDA (X Y Z)
    (FGREATERP (COND
                  (X Y)
                  (Z))
           0])
```

)

(RPAQQ **TESTNUM**
```
       ((TESTVALS (.NT1 1 2 3)
                -3.0
                (.NT2 1 2 3)
                0.0
                (.NT2 1.0 2 3)
                1.5
                (.NT3 T 3 4)
                T
                (.NT3 NIL 3 -4)
                NIL)
        (TESTPAIRS ((.LLSH . LLSH)
```

```
                    (.LRSH . LRSH)
                    (.LSH . LSH)
                    (.RSH . RSH))
              (0 3 -3 31 32 1 -1 0 -31 -32 NIL))
        (TESTPAIRS ((.ITIMES . ITIMES)
                    (.IPLUS . IPLUS)
                    (.IQUOTIENT . IQUOTIENT)
                    (.IREMAINDER . IREMAINDER)
                    (.ADD1 . ADD1)
                    (.SUB1 . SUB1)
                    (.LOGAND . LOGAND)
                    (.LOGOR . LOGOR)
                    (.LOGXOR . LOGXOR))
              (0 3 -3 2047 2048 -1 0 -1431655766 NIL))))

(DEFINEQ

(.NCONC
  [LAMBDA (X Y)                                          ; Edited 8-Apr-88 13:05 by amd
                                                         ; lmm: 19-JUN-76 20 21
    (NCONC X Y])

(..NCONC
  [LAMBDA (X Y)                                          ; Edited 8-Apr-88 13:05 by amd
                                                         ; lmm: 19-JUN-76 20 46
    (PROG NIL
          (NCONC X Y)
          (RETURN X])

(.AND
  [LAMBDA (X Y Z W)                                      ; Edited 8-Apr-88 13:05 by amd
                                                         ; lmm: 19-JUN-76 20 54
    (AND (EVAL X)
         (EVAL Y)
         (EVAL Z)
         (EVAL W])

(.FRPLNODE
  [LAMBDA (X A D)                                        ; Edited 8-Apr-88 13:05 by amd
                                                         ; lmm: 19-JUN-76 21 23
    (FRPLNODE X A D])

(.OR
  [LAMBDA (X Y Z W)                                      ; Edited 8-Apr-88 13:05 by amd
                                                         ; lmm: 19-JUN-76 21 28
    (OR (EVAL X)
        (EVAL Y)
        (EVAL Z)
        (EVAL W])

(.FRPLNODE2
  [LAMBDA (X Y)                                          ; Edited 8-Apr-88 13:06 by amd
                                                         ; lmm: 19-JUN-76 21 47
    (FRPLNODE2 X Y])

(..NCONC3
  [LAMBDA (X Y Z)                                        ; Edited 8-Apr-88 13:06 by amd
                                                         ; lmm: 19-JUN-76 23 9
    (PROG NIL
          (NCONC X Y Z)
          (RETURN X])

(.NCONC3
  [LAMBDA (X Y Z W)                                      ; Edited 8-Apr-88 13:06 by amd
                                                         ; lmm: 19-JUN-76 22 54
    (NCONC X Y Z W])

(SELECTTEST
  [LAMBDA (X)                                            ; Edited 8-Apr-88 13:06 by amd
                                                         ; lmm: 19-JUN-76 22 16
    (SELECTQ X
        (0 0)
        (1 1)
        (A 'A)
        ((B C D)
            'C)
        NIL])
```

```
(.MKLIST
  [LAMBDA (X)                                          ; Edited  8-Apr-88 13:06 by amd
                                                       ; lmm: 21-JUN-76 0 56
    (MKLIST X])

(.EQMEMB
  [LAMBDA (X Y)                                        ; Edited  8-Apr-88 13:06 by amd
                                                       ; lmm: 21-JUN-76 1 16
    (EQMEMB X Y])

(.NCONC1
  [LAMBDA (X Y)                                        ; Edited  8-Apr-88 13:06 by amd
                                                       ; lmm: 21-JUN-76 1 33
    (NCONC1 X Y])

(.GETPROPLIST
  [LAMBDA (X)                                          ; Edited  8-Apr-88 13:07 by amd
                                                       ; lmm: 21-JUN-76 2 48
    (GETPROPLIST X])

(.SETPROPLIST
  [LAMBDA (X Y)                                        ; Edited  8-Apr-88 13:07 by amd
                                                       ; lmm: 21-JUN-76 2 53
    (SETPROPLIST X Y])

(.FGETD
  [LAMBDA (X)
    (FGETD X])

(..FRPLNODE2
  [LAMBDA (L M)                                        ; Edited  8-Apr-88 13:07 by amd
    (FRPLNODE2 L M])

(!AND
  [LAMBDA (A B C)                                      ; Edited  8-Apr-88 13:07 by amd
    (LIST (AND A B)
          (COND
             ((AND A B)
              1)
             (T 2))
          (COND
             ((NOT (AND A B))
              3)
             (T 4))
          (AND (AND A B)
               5)
          (OR (AND A B)
              6)
          (PROG1 7
              (AND A B (SETQ C 9)))
          C])

(!OR
  [LAMBDA (A B C)                                      ; Edited  8-Apr-88 13:07 by amd
    (LIST (OR A B)
          (COND
             ((OR A B)
              1)
             (T 2))
          (COND
             ((NOT (OR A B))
              3)
             (T 4))
          (OR (OR A B)
              5)
          (OR (OR A B)
              6)
          (PROG1 7
              (OR A B (SETQ C 9)))
          C])

)

(RPAQQ TESTMORE
       [(SETPROPLIST 'TESTATOM '(A D))
        [TESTVALS (.NCONC (LIST 1 2 3)
                          (LIST 4 5 6))
                  '(1 2 3 4 5 6)
                  (.NCONC NIL 'A)
```

```
              'A
              (CDR NIL)
              NIL
              (..NCONC (LIST 1 2 3)
                     (LIST 4 5 6))
              '(1 2 3 4 5 6)
              (.AND T T T 3)
              3
              [.AND NIL '(PRINT '(.AND failed]
              NIL
              (.FRPLNODE (CONS)
                     2 3)
              '(2 . 3)
              (.OR 1 2 3)
              1
              (.FRPLNODE2 (CONS 'A 'B)
                     (CONS 'C 'D))
              '(C . D)
              (..FRPLNODE2 (CONS 'A 'B)
                     '(NIL))
              '(NIL)
              (.NCONC3 (LIST 1)
                     (LIST 2)
                     (LIST 3)
                     (LIST 4))
              '(1 2 3 4)
              (..NCONC3 (LIST 1)
                     (LIST 2)
                     (LIST 3)
                     (LIST 4))
              '(1 2 3)
              (LIST (SELECTTEST NIL)
                    (SELECTTEST 1)
                    (SELECTTEST 0)
                    (SELECTTEST 'A)
                    (SELECTTEST 'C))
              '(NIL 1 0 A C)
              (.MKLIST)
              NIL
              (.MKLIST 3)
              '(3)
              (.MKLIST '(3))
              '(3)
              (.EQMEMB 1 1)
              T
              (.EQMEMB 1 '(1))
              T
              (.EQMEMB 'A 'TESTATOM)
              NIL
              (.EQMEMB 'A '(D E F))
              NIL
              (.NCONC1 (LIST 1 2 3)
                     4)
              '(1 2 3 4)
              (LIST (!AND 1 2)
                    (!AND NIL 2)
                    (!AND 1 NIL)
                    (!AND NIL NIL))
              '((2 1 4 5 2 7 9)
                (NIL 2 3 NIL 6 7 NIL)
                (NIL 2 3 NIL 6 7 NIL)
                (NIL 2 3 NIL 6 7 NIL))
              (LIST (!OR 1 2)
                    (!OR NIL 2)
                    (!OR 1 NIL)
                    (!OR NIL NIL))
              '((1 1 4 1 1 7 NIL)
                (2 1 4 2 2 7 NIL)
                (1 1 4 1 1 7 NIL)
                (NIL 2 3 5 6 7 9]
       (SETPROPLIST 'TESTATOM '(A B C D E F))
       (TESTIFSAME (.SETPROPLIST SETPROPLIST 'TESTATOM '(D E F G))
              (.GETPROPLIST GETPROPLIST 'TESTATOM)
              (.FGETD FGETD 'CONS])

(DEFINEQ

(!ADD1VAR
  [NLAMBDA (ADD1X)                                    ; Edited 8-Apr-88 13:07 by amd
    ;; COMPILES OPEN
    (DECLARE (LOCALVARS . T))
    (SET ADD1X (ADD1 (EVALV ADD1X])

(!APPEND
  [LAMBDA L                                           ; Edited 8-Apr-88 13:08 by amd
```

```
                                                                    ; lmm: 2-JUL-76 4 3
    (SELECTQ L
         (0 NIL)
         (1 (APPEND2 (ARG L 1)
                     NIL))
         (2 (APPEND2 (ARG L 1)
                     (ARG L 2)))
         (PROG ((V (ARG L L))
                (I L))
           LP   (COND
                   ((ZEROP (SETQ I (SUB1 I)))
                    (RETURN V))
                   (T (SETQ V (APPEND2 (ARG L I)
                                       V))
                      (GO LP])
```

(APPEND2

```
  [LAMBDA (X Y)                                                     ; Edited  8-Apr-88 13:08 by amd
                                                                    ; lmm: 2-JUL-76 4 1
    (COND
       ((NLISTP X)
        Y)
       (T (CONS (CAR X)
                (APPEND2 (CDR X)
                         Y])
```

(!ASSOC

```
  [LAMBDA (KEY LST)                                                 ; Edited  8-Apr-88 13:08 by amd
                                                                    ; lmm: 6-JUL-76 20 11
    ;; BYTECODE
    (COND
       ((NLISTP LST)
        NIL)
       ((EQ KEY (CAAR LST))
        (CAR LST))
       (T (!ASSOC KEY (CDR LST])
```

(!ATTACH

```
  [LAMBDA (X LST)                                                   ; Edited  8-Apr-88 13:09 by amd
    ;; MSOPVAL COPY CAR COPY1 CDR CONS RPLACD SWAP FRPLACA
    (RPLNODE LST X (CONS (CAR LST)
                         (CDR LST])
```

(!CHANGEPROP

```
  [LAMBDA (X PROP1 PROP2)                                           ; Edited  8-Apr-88 13:09 by amd
    ;; FMEMB !!! UGH
    (COND
       ((SETQ PROP1 (FMEMB PROP1 (GETPROPLIST X)))
        (FRPLACA PROP1 PROP2)
        X])
```

(!COPY

```
  [LAMBDA (X)                                                       ; Edited  8-Apr-88 13:09 by amd
                                                                    ; lmm: 6-JUL-76 20 12
    ;; COLLECT?
    (COND
       ((NLISTP X)
        X)
       (T (CONS (!COPY (CAR X))
                (!COPY (CDR X])
```

(!DEFLIST

```
  [LAMBDA (L PROP)                                                  ; Edited  8-Apr-88 13:09 by amd
                                                                    ; lmm: 6-JUL-76 20 12
    (COND
       ((NLISTP L)
        NIL)
       (T (PUTPROP (CAAR L)
                   PROP
                   (CADAR L))
          (!DEFLIST (CDR L)
                    PROP])
```

(!DREMOVE

```
  [LAMBDA (X L)                                                     ; Edited  8-Apr-88 13:10 by amd
                                                                    ; lmm: 6-JUL-76 20 12
```

```
    (COND
       ((NLISTP L)
        NIL)
       [(EQ X (CAR L))
        (COND
           ((CDR L)
            (!DREMOVE X (FRPLNODE L (CADR L)
                                    (CDDR L]
       (T ;; GET RID OF PROG AND RECURSE?
          (PROG (Z)
                (DECLARE (LOCALVARS Z))
                (SETQ Z L)
            LP  [COND
                   ((NLISTP (CDR L))
                    (RETURN Z))
                   ((EQ X (CADR L))
                    (FRPLACD L (CDDR L)))
                   (T (SETQ L (CDR L]
                (GO LP])
```

(!DREVERSE

```
  [LAMBDA (L)
    (DREV L NIL])
```

(DREV

```
  [LAMBDA (L Z)                                    ; Edited  8-Apr-88 13:10 by amd
    (PROG (Y)
      R1  (COND
             ((NLISTP (SETQ Y L))
              (RETURN Z)))
          (SETQ L (CDR L))
          (SETQ Z (FRPLACD Y Z))
          (GO R1])
```

(!DSUBST

```
  [LAMBDA (X Y Z)                                  ; Edited  8-Apr-88 13:10 by amd
                                                   ; lmm: 2-JUL-76 17 20
    (COND
       ((EQ Y Z)
        (COPY X))
       ((NLISTP Z)
        Z)
       (T [COND
             ((EQUAL Y (CAR Z))
              (FRPLACA Z (COPY X)))
             (T (!DSUBST X Y (CAR Z]
          (COND
             ((AND Y (EQUAL Y (CDR Z)))
              (FRPLACD Z (COPY X)))
             (T (!DSUBST X Y (CDR Z))
                Z])
```

(!EVERY

```
  [LAMBDA (EVERYX EVERYFN1 EVERYFN2)
    (COND
       ((NLISTP EVERYX)
        T)
       ((NULL (APPLY* EVERYFN1 (CAR EVERYX)
                      EVERYX))
        NIL)
       (T (!EVERY (APPLY* (OR EVERYFN2 'CDR)
                          EVERYX)
                  EVERYFN1 EVERYFN2])
```

(!GETP

```
  [LAMBDA (ATM PROP)                               ; Edited  8-Apr-88 13:11 by amd
    ;; HAS BYTE CODE
    (AND (LITATOM ATM)
         (PROG NIL
               (SETQ ATM (GETPROPLIST ATM))
          LOOP
               [COND
                  ((OR (NLISTP ATM)
                       (NLISTP (CDR ATM)))
                   (RETURN NIL))
                  ((EQ (CAR ATM)
                       PROP)
                   (RETURN (CADR ATM]
               (SETQ ATM (CDDR ATM))
               (GO LOOP])
```

(!INTERSECTION

```
[LAMBDA (X Y)                                          ; Edited  8-Apr-88 13:11 by amd
                                                       ; lmm: 6-JUL-76 20 8

   (AND (LISTP X)
        (COND
           ([AND (MEMBER (CAR X)
                         Y)
                 (NOT (MEMBER (CAR X)
                              (CDR X]
            (CONS (CAR X)
                  (!INTERSECTION (CDR X)
                                 Y)))
           (T (!INTERSECTION (CDR X)
                             Y])
```

(!LAST

```
[LAMBDA (X)                                            ; Edited  8-Apr-88 13:11 by amd
   ;; BYTE CODE
   (COND
      ((NLISTP X)
       X)
      ((NLISTP (CDR X))
       X)
      (T (!LAST (CDR X])
```

(!LASTN

```
[LAMBDA (L N)                                          ; Edited  8-Apr-88 13:38 by amd
   ;; UGH! NCONC1 SHOULD BE COLLECT
   (AND (LISTP L)
        (PROG ((X (FNTH L N))
               Y)
              (COND
                 ((NULL X)
                  (RETURN)))
          LP  [COND
                 ((NULL (SETQ X (CDR X)))
                  (RETURN (CONS Y L]
              (SETQ Y (NCONC1 Y (CAR L)))
              (SETQ L (CDR L))
              (GO LP])
```

(!LDIFF

```
[LAMBDA (X Y Z)                                        ; Edited  8-Apr-88 13:12 by amd
                                                       ; lmm: 6-JUL-76 20 10

   (COND
      ((EQ X Y)
       Z)
      (Z (NCONC Z (!LDIFF X Y)))
      ((NULL Y)
       X)
      ((NLISTP X)
       (ERROR '"LDIFF: not a tail" Y))
      (T (CONS (CAR X)
               (!LDIFF (CDR X)
                       Y])
```

(!LENGTH

```
[LAMBDA (L)                                            ; Edited  8-Apr-88 13:12 by amd
   ;; BYTE CODE
   (COND
      ((NLISTP L)
       0)
      (T (ADD1 (LENGTH (CDR L])
```

(!LISTGET

```
[LAMBDA (LST PROP)                                     ; Edited  8-Apr-88 13:12 by amd
                                                       ; lmm: 6-JUL-76 20 13

   ;; BYTE CODE
   ;; like getp but works on lists, searching them two cdrs at a time.
   (AND (LISTP LST)
        (COND
           ((EQ PROP (CAR LST))
            (CADR LST))
           (T (!LISTGET (CDDR LST)
                        PROP])
```

(!LSUBST

```
[LAMBDA (X Y Z)                                          ; Edited  8-Apr-88 13:12 by amd
                                                         ; lmm: 2-JUL-76 16 25
```

;; Substitutes X as a segment for Y in Z. E.g. !LSUBST ((A B) Y (X Y Z)) is (X A B Z) not meaningful for Y an atom and CDR of a list. if X is NIL, ;; operation effectively deletes Y, i.e. produces a copy without Y in it.

```
(COND
   ((NULL Z)
    NIL)
   ((NLISTP Z)
    (COND
       ((EQ Y Z)
        X)
       (T Z)))
   [(EQUAL Y (CAR Z))
    (NCONC (COPY X)
           (!LSUBST X Y (CDR Z]
   (T (CONS (!LSUBST X Y (CAR Z))
            (!LSUBST X Y (CDR Z])
```

(!MAP

```
[LAMBDA (MAPX MAPFN1 MAPFN2)                             ; Edited  8-Apr-88 13:13 by amd
                                                         ; lmm: 6-JUL-76 20 5
  (COND
     ((NLISTP MAPX)
      NIL)
     (T (APPLY* MAPFN1 MAPX)
        (!MAP (COND
                 (MAPFN2 (APPLY* MAPFN2 MAPX))
                 (T (CDR MAPX)))
              MAPFN1 MAPFN2])
```

(!GET

```
[LAMBDA (LST PROP)                                       ; Edited  8-Apr-88 13:13 by amd
```

;; GIVE IT A MACRO

```
  (CADR (MEMB PROP LST])
```

(!GETLIS

```
[LAMBDA (X PROPS)
  (PROG [(Z (COND
               ((LITATOM X)
                (GETPROPLIST X))
               (T X]
    LP  (COND
           ((OR (NLISTP Z)
                (FMEMB (CAR Z)
                       PROPS))
            (RETURN Z)))
        (SETQ Z (CDR Z))
        (GO LP])
```

(!MEMB

```
[LAMBDA (X Y)                                            ; Edited  8-Apr-88 13:13 by amd
                                                         ; lmm: 6-JUL-76 20 15
  (COND
     ((NLISTP Y)
      NIL)
     ((EQ X (CAR Y))
      Y)
     (T (!MEMB X (CDR Y])
```

(!NTH

```
[LAMBDA (X N)
  (COND
     ((IGREATERP 1 N)
      (CONS NIL X))
     (T (PROG NIL
          LP  (COND
                 ((NOT (IGREATERP N 1))
                  (RETURN X))
                 ((NLISTP X)
                  (RETURN NIL)))
              (SETQ X (CDR X))
              (SETQ N (SUB1 N))
              (GO LP])
```

(.COLLCT

```
[LAMBDA (X Y)                                            (* lmm%: 28-JUN-76 12 45)
  (COLLCT X Y])
```

```
(.ENDCOLLCT
  [LAMBDA (Y)                                                    (* lmm%: 28-JUN-76 12 44)
    (ENDCOLLCT Y])

(MYAPPEND1
  [LAMBDA (X Y)                                                  (* lmm%: "14-AUG-76 22:32:20")
    (COND
       ((NLISTP X)
        Y)
       (T (CONS (CAR X)
                (MYAPPEND1 (CDR X)
                       Y])

(MYAPPEND2
  [LAMBDA (X Y)                                                  (* lmm%: 30-JUN-76 16 57)
    (PROG (V)
      LP  (COND
             ((NLISTP X)
              (RETURN (ENDCOLLCT V Y)))
             (T (SETQ V (COLLCT V (CAR X)))
                (SETQ X (CDR X))
                (GO LP])

(COLLCT
  [LAMBDA (LST NEWITEM)                                          (* lmm%: 28-JUN-76 10 47)
    (COND
       ((NULL LST)
        (RPLACD (SETQ LST (LIST NEWITEM))
               LST))
       (T (CDR (RPLACD LST (CONS NEWITEM (CDR LST])

(ENDCOLLCT
  [LAMBDA (X Y)                                                  (* lmm%: 30-JUN-76 16 19)
    (PROG1 (CDR X)
           (FRPLACD X Y])

(.ATTACH
  [LAMBDA (X Y)                                                  (* lmm%: 28-JUN-76 13 10)
    (ATTACH X Y])

(.APPEND0
  [LAMBDA NIL                                                    (* lmm%: 2-JUL-76 14 22)
    (APPEND])

(.APPEND1
  [LAMBDA (X)                                                    ; Edited  8-Apr-88 13:13 by amd
                                                                 ; lmm: 2-JUL-76 14 23
    (APPEND X])

(.APPEND2
  [LAMBDA (X Y)                                                  ; Edited  8-Apr-88 13:13 by amd
                                                                 ; lmm: 2-JUL-76 14 23
    (APPEND X Y])

(.APPEND3
  [LAMBDA (X Y Z)                                                ; Edited  8-Apr-88 13:14 by amd
                                                                 ; lmm: 2-JUL-76 14 23
    (APPEND X Y Z])

(.APPEND4
  [LAMBDA (X Y Z W)                                              ; Edited  8-Apr-88 13:14 by amd
                                                                 ; lmm: 2-JUL-76 14 23
    (APPEND X Y Z W])

(.MAPCGETP
  [LAMBDA (AT PROP FN)                                           ; Edited  8-Apr-88 13:14 by amd
                                                                 ; lmm: 7-JUL-76 1 22
    (MAPC (GETP AT PROP)
          (FUNCTION (LAMBDA (X)
                      (APPLY* FN X])

)

(RPAQQ TESTMISC
```

```
((TESTVALS (PROGN (SETQ TESTATOM 3)
                  (!ADD1VAR TESTATOM)
                  TESTATOM)
        4
        (!APPEND '(A B C)
               '(D E F)
               '(G H I))
        '(A B C D E F G H I)
        (!APPEND '(A B C))
        '(A B C)
        [!ASSOC 0 '((3 . 2)
                    (0 . 1]
        '(0 . 1)
        [!ASSOC 3 '((1 . 2)
                    (2 . 3]
        NIL
        (!ATTACH NIL T)
        '(:ERROR "T is not a LIST.")
        (!ATTACH 'A (LIST 1 2 3))
        '(A 1 2 3)
        [PROGN (SETPROPLIST 'TESTATOM (LIST 1 2 3 4))
               (!CHANGEPROP 'TESTATOM '3 '10)
               (LIST (!GETP 'TESTATOM 3)
                     (!GETP 'TESTATOM 10)
                     (!GETP 'TESTATOM 1)
                     (GETPROPLIST 'TESTATOM]
        '(NIL 4 2 (1 2 10 4))
        (!COPY '(A 1 1.3 "FOO" . XX))
        '(A 1 1.3 "FOO" . XX)
        [!COPY '((A . 3)
                 (B C 10 3 . 10]
        '((A . 3)
          (B C 10 3 . 10))
        (!DREMOVE 3 (LIST 3 1 3 5 7))
        '(1 5 7)
        [!DSUBST 3 10 (!COPY '((A . 3)
                               (B C 10 3 . 10)
                               (10 . A) . A]
        '((A . 3)
          (B C 3 3 . 3)
          (3 . A) . A)
        (!EVERY '(1 NIL 2 NIL 4 NIL)
               'SMALLP
               'CDDR)
        T
        (!EVERY '(A B C . 3)
               'LITATOM)
        T
        (!EVERY '(1 2 3 A)
               'SMALLP)
        NIL
        (!GET '(A B C D E)
              'A)
        'B
        (!GET '(A B C D E)
              'B)
        'C
        (!GETLIS '(A B C D E)
               '(1 3 B))
        '(B C D E)
        [PROGN (!DEFLIST '((FOO FIE)
                           (FUM FEE))
                         'PROPNAME)
               (LIST (GETP 'FOO 'PROPNAME)
                     (GETP 'FUM 'PROPNAME]
        '(FIE FEE)
        (LIST (!DREVERSE (SETQ A (LIST 1 2 3 4 5)))
              A)
        '((5 4 3 2 1)
          (1))
        (!GET '(A B C . D)
              'A)
        'B
        (!INTERSECTION '(1 3 2 4)
               '(4 2 1))
        '(1 2 4)
        [LIST (!LAST 3)
              (!LAST '(A . B))
              (!LAST '(A B C D E F G H I J K L M N O P Q R S T U V W X Y Z 1 2 3 4 5 6 7 8 9 0 ! @ %# $
                         %% ~ & * %( %) - = + \ %| %[ %] { } _ ^ %: ; %' %" %, < > %. ? / END]
        '(3 (A . B)
            (END))
        (!LASTN (LIST 1 2 3 4 5)
               1)
        '((1 2 3 4)
          5)
        (!LASTN (LIST 1 2 3)
```

```
                0)
          '((1 2 3))
          (!LDIFF (SETQ A '(A B C D E F G H))
                  (!NTH A 4))
          '(A B C)
          (!LENGTH T)
          0
          (!LENGTH '(A B C D E F))
          6
          (!LISTGET '(A B C D)
                 'B)
          NIL
          (!LISTGET '(A B C D E)
                 'A)
          'B
          [!LSUBST '(A B C)
                 'D
                 (!COPY '((3 . D)
                          (D . 3)
                          (X D Y]
          '((3 A B C)
            (A B C . 3)
            (X A B C Y))
          (PROG ((V 0))
                (RETURN (LIST [!MAP '(1 2 3 4)
                                    (FUNCTION (LAMBDA (X)
                                                      (SETQ V (IPLUS V (CAR X]
                              V)))
          '(NIL 10))
      (TESTVALS (ENDCOLLCT (COLLCT (COLLCT (COLLCT (COLLCT NIL 'A)
                                                   10)
                                   14))
          '(A 10 14)
          (ENDCOLLCT (.COLLCT (.COLLCT (.COLLCT (.COLLCT NIL 'A)
                                                10)
                              14))
          '(A 10 14)
          (.ENDCOLLCT (.COLLCT (.COLLCT (.COLLCT (.COLLCT NIL 'A)
                                                 10)
                               14))
          '(A 10 14)
          (MYAPPEND1 '(A B C)
                 '(D E F))
          '(A B C D E F)
          (MYAPPEND2 (LIST 1 2 3)
                  (LIST 4 5 6))
          '(1 2 3 4 5 6)
          (.ATTACH 1 (LIST 2 3 4 5))
          '(1 2 3 4 5)
          (.APPEND1 '(A B C))
          '(A B C)
          (.APPEND2 '(A B C)
                 '(D E F))
          '(A B C D E F)
          (.APPEND3 '(A B)
                 '(C D)
                 '(E F))
          '(A B C D E F)
          (.APPEND4 '(A B)
                 NIL
                 '(C D)
                 NIL)
          '(A B C D)
          (PROG ((S 0))
                (PUT 'A 'B '(1 2 3 4 5))
                [.MAPCGETP 'A 'B (FUNCTION (LAMBDA (X)
                                                   (SETQ S (IPLUS X S]
                (RETURN S))
          15)))

(DEFINEQ

(.MAP
  [LAMBDA (X Y Z)                                          ; Edited 8-Apr-88 13:14 by amd
    (LIST 1 (PROG ((C 0))
                  [MAP X (FUNCTION (LAMBDA (Y)
                                   (SETQ C (IPLUS (LENGTH Y)
                                                  C]
                  (RETURN C))
          7])

(.MAPC
  [LAMBDA (X Y Z)
    (LIST 1 (PROG ((C 0))
                  [MAPC X (FUNCTION (LAMBDA (Y)
                                    (SETQ C (IPLUS Y C]
```

```
                (RETURN C))
          (COND
             ((PROG ((C 0))
                    [MAPC X (FUNCTION (LAMBDA (Y)
                                      (SETQ C (IPLUS Y C]
                    (RETURN C))
              2)
             ((NOT (PROG ((C 0))
                         [MAPC X (FUNCTION (LAMBDA (Y)
                                           (SETQ C (IPLUS Y C]
                         (RETURN C)))
              3)
             (T 4))
          (OR (PROG ((C 0))
                    [MAPC X (FUNCTION (LAMBDA (Y)
                                      (SETQ C (IPLUS Y C]
                    (RETURN C))
              5)
          (AND (PROG ((C 0))
                     [MAPC X (FUNCTION (LAMBDA (Y)
                                       (SETQ C (IPLUS Y C]
                     (RETURN C))
               6)
          (PROGN (PROG ((C 0))
                       [MAPC X (FUNCTION (LAMBDA (Y)
                                         (SETQ C (IPLUS Y C]
                       (RETURN C))
                 7])
```

(**.MAPCEFF**

```
  [LAMBDA (X Y Z)                                          ; Edited  8-Apr-88 13:14 by amd
    (LIST 1 (PROGN (PROG ((C 0))
                         [MAPC X (FUNCTION (LAMBDA (Y)
                                           (SETQ C (IPLUS Y C]
                         (RETURN C))
                   7])
```

)

```
(RPAQQ TESTMAP
       [(TESTVALS (.MAP)
              '(1 0 7)
              (.MAP '(1 2 3 4 5))
              '(1 15 7))
        (TESTVALS (.MAPC '(1 2 3 4))
              '(1 10 2 10 6 7)
              (.MAPCEFF)
              '(1 7])

(DEFINEQ
```

(**.FPLUS**

```
  [LAMBDA (X Y)                                            ; Edited  8-Apr-88 13:14 by amd
                                                           ; lmm: 22-JUN-76 0 39
    ;; subr
    (FPLUS X Y])
```

(**.FTIMES**

```
  [LAMBDA (X Y)                                            ; Edited  8-Apr-88 13:15 by amd
                                                           ; lmm: 22-JUN-76 0 40
    ;; SUBR*
    (FTIMES X Y])
```

(**.FDIFFERENCE**

```
  [LAMBDA (X Y)                                            ; Edited  8-Apr-88 13:15 by amd
                                                           ; lmm: 22-JUN-76 0 40
    ;; CEXPR
    (FDIFFERENCE X Y])
```

(**.FQUOTIENT**

```
  [LAMBDA (X Y)                                            ; Edited  8-Apr-88 13:15 by amd
                                                           ; lmm: 22-JUN-76 0 40
    (FQUOTIENT X Y])

)

(RPAQQ TESTLINK ((TESTIFSAME (.FPLUS FPLUS 2 10)
                        (.FTIMES FTIMES 3 5)
                        (.FDIFFERENCE FDIFFERENCE 4.5 7.9)
                        (.FQUOTIENT FQUOTIENT 4.5 10.3))))
```

```
(DEFINEQ

(NONLOCALGO
  [LAMBDA (A B C)                                          ; Edited  8-Apr-88 13:15 by amd
                                                           ; lmm: 28-JUN-76 14 9

    (LIST -2 -1 (PROG ((C 1)
                       (D 2)
                       (E 3)
                       F)
                      (SETQ F (LIST C D E))
                      [RETURN (PROG ((G 13)
                                     (H 14)
                                     (I 15))
                                    (RETURN (LIST G H I (PROG ((I H)
                                                               (H I))
                                                              (COND
                                                                 (F (GO POPOUT)))
                                                              (RETURN (LIST I H]
                  POPOUT
                      (RETURN (LIST D E F])

(CNTDWN
  [LAMBDA (X)                                              ; Edited  8-Apr-88 13:15 by amd
                                                           ; lmm: 21-JUN-76 3 54

    (PROG NIL
      LP  (SELECTQ X
                (0 (RETURN))
                (PROGN (SETQ X (SUB1 X))
                       (GO LP])

(JUMPAROUND
  [LAMBDA NIL                                              ; Edited  8-Apr-88 13:15 by amd
                                                           ; lmm: 22-JUN-76 1 23

    (COND
       (NIL T)
       (T NIL])

)

(RPAQQ TESTJUMP ([TESTVALS (NONLOCALGO)
                           '(-2 -1 (2 3 (1 2 3]
                 (TESTVALS (CNTDWN 0)
                           NIL)
                 (TESTVALS (CNTDWN 20)
                           NIL)))

(DEFINEQ

(FN2
  [LAMBDA (X Y)                                            ; Edited  8-Apr-88 13:16 by amd
                                                           ; lmm: 19-JUN-76 2 6

    (LIST3 X Y])

(FN3
  [LAMBDA (X Y Z)                                          ; Edited  8-Apr-88 13:16 by amd
                                                           ; lmm: 19-JUN-76 1 53

    (LIST3 X Y Z])

(FN1
  [LAMBDA (X)                                              ; Edited  8-Apr-88 13:16 by amd
                                                           ; lmm: 19-JUN-76 2 8

    (LIST3 X])

(.IVAR
  [LAMBDA (A B C D E F G H I J K L M N O)                  ; Edited  8-Apr-88 13:16 by amd
                                                           ; lmm: 19-JUN-76 2 12

    (IVAR A B C D E F G H I J K L M N O])

)

(RPAQQ TESTFNX
       [(TESTVALS (FN3 1 2 3)
                  '(1 2 3)
                  (FN2 1 2)
                  '(1 2 NIL)
                  (FN1 1)
                  '(1 NIL NIL)
                  (.IVAR 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15)
                  '(1 2 3 4 5 6 7 8 9 10 11 12 13 14 15])

(DEFINEQ
```

```
(.GETP
  [LAMBDA (X Y)
    (GETP X Y])

(.GETPROP
  [LAMBDA (X Y)
    (GETPROP X Y])

(.RPLACA
  [LAMBDA (X Y)                                          ; Edited  8-Apr-88 13:17 by amd
                                                         ; lmm: 18-JUN-76 14 22
    (RPLACA X Y])

(.RPLACD
  [LAMBDA (X Y)                                          ; Edited  8-Apr-88 13:17 by amd
                                                         ; lmm: 18-JUN-76 14 22
    (RPLACD X Y])

(.FRPLACA
  [LAMBDA (X Y)                                          ; Edited  8-Apr-88 13:17 by amd
                                                         ; lmm: 18-JUN-76 14 46
    (FRPLACA X Y])

(.GET
  [LAMBDA (X Y)                                          ; Edited  8-Apr-88 13:17 by amd
    (LISTGET X Y])

(.FRPLACD
  [LAMBDA (X Y)                                          ; Edited  8-Apr-88 13:17 by amd
                                                         ; lmm: 18-JUN-76 14 47
    (FRPLACD X Y])

(.ASSOC
  [LAMBDA (X Y)
    (ASSOC X Y])

(.LENGTH
  [LAMBDA (X)
    (LENGTH X])

(.LAST
  [LAMBDA (X)
    (LAST X])

(.GETHASH
  [LAMBDA (X Y)
    (GETHASH X Y])

(.FMEMB
  [LAMBDA (X Y)
    (FMEMB X Y])

)

(RPAQQ TESTFN
       [[SETPROPLIST (PUTHASH 1 'TESTATOM (SETQ TARRAY (HARRAY 10)))
              (PUTHASH 3 '(A (B)
                             C D E F G (H]
        (SETQQ LONGLIST TESTATOM)
        (RPTQ 2048 (SETQ LONGLIST (CONS T LONGLIST)))
        (TESTIFSAME (.GETP GETP 'TESTATOM 'A)
              (.GETPROP GETP 'TESTATOM 'B)
              (.GETP GETP 3 NIL)
              (.RPLACA RPLACA (CONS 'A 'B)
                     'C)
              (.RPLACA RPLACA NIL T)
              (.RPLACA RPLACA NIL NIL)
              (.RPLACA RPLACA "foo" "fum")
              (.RPLACD RPLACD (CONS 'A 'B)
                     'C)
              (.FRPLACA FRPLACA (CONS 'A 'B)
                     'C)
              (.FRPLACD FRPLACD (CONS 'A 'B)
                     'C)
              (.GET LISTGET '(A B C . 3)
```

```
                'C)
          (.GET LISTGET '(A B C . TESTATOM)
                'A)
          (.GET LISTGET '(A B C . TESTATOM)
                'D)
          (.GET LISTGET 'TESTATOM 'D)
          (.LENGTH LENGTH '(1 3 . 4))
          (.LENGTH LENGTH LONGLIST)
          (.ASSOC ASSOC '((A . B)
                          (C . D))
                 'A)
          (.ASSOC ASSOC 'TESTATOM 'H)
          (.LAST LAST LONGLIST)
          (.GETHASH GETHASH 3)
          (.GETHASH GETHASH 1 TARRAY)
          (.FGETD FGETD 'RECLAIM)
          (.FGETD FGETD '.FGETD)
          (.FMEMB FMEMB 'A '(D B C A))
          (.FMEMB FMEMB 'A '(D B C])
```

```
(RPAQQ TESTEDIT
       [(TESTVALS [EDITE (LIST 1 2 3 4)
                         '((1)
                           (N 5)
                           (2 (F (G H)))
                           F F (SW 1 2)
                           ^ F H !0 (1 P]
                  '(2 ((P H)
                       F)
                      4 5])
```

```
(DEFINEQ

(.CONS
  [LAMBDA (A B)
    (CONS A B])

(LIST0
  [LAMBDA NIL                                          ; Edited  8-Apr-88 13:18 by amd
                                                       ; lmm: 21-JUN-76 15 57

    (LIST])

(LIST1
  [LAMBDA (X)
    (LIST X])

(LIST2
  [LAMBDA (A B)
    (LIST A B])

(LIST3
  [LAMBDA (X Y Z)
    (LIST X Y Z])

(LIST4
  [LAMBDA (A B C D)                                    ; Edited  8-Apr-88 13:18 by amd
                                                       ; lmm: 21-JUN-76 15 56

    (LIST A B C D])

(LIST5
  [LAMBDA (A B C D E)                                  ; Edited  8-Apr-88 13:18 by amd
                                                       ; lmm: 21-JUN-76 15 56

    (LIST A B C D E])

(LIST6
  [LAMBDA (A B C D E F)                                ; Edited  8-Apr-88 13:18 by amd
                                                       ; lmm: 21-JUN-76 15 57

    (LIST A B C D E F])

(LIST7
  [LAMBDA (A B C D E F G)                              ; Edited  8-Apr-88 13:18 by amd
                                                       ; lmm: 21-JUN-76 15 57

    (LIST A B C D E F G])

(LIST8
  [LAMBDA (A B C D E F G H)                            ; Edited  8-Apr-88 13:19 by amd
                                                       ; lmm: 21-JUN-76 15 57

    (LIST A B C D E F G H])
```

```
(LIST9
  [LAMBDA (A B C D E F G H I)                          ; Edited  8-Apr-88 13:19 by amd
                                                       ; lmm: 21-JUN-76 15 57
    (LIST A B C D E F G H I])

(LIST10
  [LAMBDA (A B C D E F G H I J)                        ; Edited  8-Apr-88 13:19 by amd
                                                       ; lmm: 25-JUN-76 5 18
    (LIST A B C D E F G H I J])

(LIST11
  [LAMBDA (A B C D E F G H I J K)                       ; Edited  8-Apr-88 13:19 by amd
                                                       ; lmm: 25-JUN-76 5 19
    (LIST A B C D E F G H I J K])

(LIST12
  [LAMBDA (A B C D E F G H I J K L)                     ; Edited  8-Apr-88 13:19 by amd
                                                       ; lmm: 28-JUN-76 8 3
    (LIST A B C D E F G H I J K L])

(LIST13
  [LAMBDA (A B C D E F G H I J K L M)                   ; Edited  8-Apr-88 13:19 by amd
                                                       ; lmm: 28-JUN-76 8 3
    (LIST A B C D E F G H I J K L M])

)

(RPAQQ TESTCONS
       [(TESTVALS (.CONS 10 12)
                  '(10 . 12)
                  (LIST0)
                  NIL
                  (LIST1 1)
                  '(1)
                  (LIST2 1 2)
                  '(1 2)
                  (LIST3 1 2 3)
                  '(1 2 3)
                  (LIST4 1 2 3 4)
                  '(1 2 3 4)
                  (LIST5 1 2 3 4 5)
                  '(1 2 3 4 5)
                  (LIST6 1 2 3 4 5 6)
                  '(1 2 3 4 5 6)
                  (LIST7 1 2 3 4 5 6 7)
                  '(1 2 3 4 5 6 7)
                  (LIST8 1 2 3 4 5 6 7 8)
                  '(1 2 3 4 5 6 7 8)
                  (LIST9 1 2 3 4 5 6 7 8 9)
                  '(1 2 3 4 5 6 7 8 9)
                  (LIST10 10 9 8 7 6 5 4 3 2 1)
                  '(10 9 8 7 6 5 4 3 2 1)
                  (LIST11 1 2 6 3 7 4 8 5 9 10 11)
                  '(1 2 6 3 7 4 8 5 9 10 11])

(DEFINEQ

(.CAR
  [LAMBDA (X)
    (CAR X])

(.CDR
  [LAMBDA (X)
    (CDR X])

(.CAAR
  [LAMBDA (X)
    (CAAR X])

(.CDAR
  [LAMBDA (X)
    (CDAR X])

(.CADR
  [LAMBDA (X)
    (CADR X])
```

```
(.CDDR
  [LAMBDA (X)
    (CDDR X])


(.CAAAAR
  [LAMBDA (X)                                          ; Edited  8-Apr-88 13:20 by amd
                                                       ; lmm: 19-JUN-76 20 57
    (CAR (CAAR (CAR X])

)

(RPAQQ TESTCAR
       ((TESTVALS (.CAR '(A))
                'A
                (.CDR '(A . B))
                'B
                [.CAAR '((A]
                'A
                [.CDAR '((A . B]
                'B
                (.CADR '(A B))
                'B
                (.CDDR '(A B . C))
                'C
                [.CAAAAR '((((A]
                'A)))

(DEFINEQ

(.LAM0
  [LAMBDA NIL
    (LAM0])


(..LAM0
  [LAMBDA NIL
    (LAM0 T T T T T T])


(.LAM1
  [LAMBDA NIL
    (LAM1])


(..LAM1
  [LAMBDA NIL
    (LAM1 -372 "extra args" "should be ignored"])


(.NLAML
  [LAMBDA NIL
    (NLAML the NLAMBDA bit should make no difference])


(LAM1LOC
  [LAMBDA (X)
    (DECLARE (LOCALVARS X))
    X])

)

(RPAQQ TESTCALL
       ((TESTVALS (LAM0)
                NIL
                (.LAM0)
                NIL
                (LAM0 T T T T T)
                NIL
                (..LAM0)
                NIL
                (LAM1)
                NIL
                (.LAM1)
                NIL
                (LAM1 -372 "extra args" "should be ignored")
                -372
                (..LAM1)
                -372
                (NLAML the NLAMBDA bit should make no difference)
                'the
                (.NLAML)
                'the
                (LAM1LOC)
                NIL
                (LAM1LOC 341)
```

```
              341
              (LAM1LOC 27 28 29)
              27)))

(DEFINEQ
```

(.PROGS

```
  [LAMBDA (X Y Z)                                          ; Edited  8-Apr-88 13:21 by amd
    (LIST 1 [PROG (X (Y (CDR Y)))
                  LPX (RETURN (LIST 3 (COND
                                      (X (SETQ Z X)
                                         (SETQ X)
                                         (GO LPX))
                                      (Y (SETQ Y NIL)
                                         (RETURN 17))
                                      (T (LIST Z]
            (COND
               ([PROG (X (Y (CDR Y)))
                      LPX (RETURN (LIST 3 (COND
                                          (X (SETQ Z X)
                                             (SETQ X)
                                             (GO LPX))
                                          (Y (SETQ Y NIL)
                                             (RETURN 17))
                                          (T (LIST Z]
                2)
               ([NOT (PROG (X (Y (CDR Y)))
                           LPX (RETURN (LIST 3 (COND
                                               (X (SETQ Z X)
                                                  (SETQ X)
                                                  (GO LPX))
                                               (Y (SETQ Y NIL)
                                                  (RETURN 17))
                                               (T (LIST Z]
                3)
               (T 4))
            (OR [PROG (X (Y (CDR Y)))
                      LPX (RETURN (LIST 3 (COND
                                          (X (SETQ Z X)
                                             (SETQ X)
                                             (GO LPX))
                                          (Y (SETQ Y NIL)
                                             (RETURN 17))
                                          (T (LIST Z]
                5)
            (AND [PROG (X (Y (CDR Y)))
                       LPX (RETURN (LIST 3 (COND
                                           (X (SETQ Z X)
                                              (SETQ X)
                                              (GO LPX))
                                           (Y (SETQ Y NIL)
                                              (RETURN 17))
                                           (T (LIST Z]
                 6)
            (PROGN [PROG (X (Y (CDR Y)))
                         LPX (RETURN (LIST 3 (COND
                                             (X (SETQ Z X)
                                                (SETQ X)
                                                (GO LPX))
                                             (Y (SETQ Y NIL)
                                                (RETURN 17))
                                             (T (LIST Z]
                   7])
```

(.SPEC

```
  [LAMBDA (X Y Z)
    (LIST 1 [IDIFFERENCE (CONSCOUNT)
                    (PROG (LISPXHIST)
                          (DECLARE (LOCALVARS . T))
                          (/RPLACA X NIL)
                          (RETURN (CONSCOUNT]
            (COND
               ([IDIFFERENCE (CONSCOUNT)
                        (PROG (LISPXHIST)
                              (DECLARE (LOCALVARS . T))
                              (/RPLACA X NIL)
                              (RETURN (CONSCOUNT]
                2)
               ([NOT (IDIFFERENCE (CONSCOUNT)
                                  (PROG (LISPXHIST)
                                        (DECLARE (LOCALVARS . T))
                                        (/RPLACA X NIL)
                                        (RETURN (CONSCOUNT]
                3)
               (T 4))
            (OR [IDIFFERENCE (CONSCOUNT)
```

```
                    (PROG (LISPXHIST)
                          (DECLARE (LOCALVARS . T))
                          (/RPLACA X NIL)
                          (RETURN (CONSCOUNT]
                5)
          (AND [IDIFFERENCE (CONSCOUNT)
                       (PROG (LISPXHIST)
                             (DECLARE (LOCALVARS . T))
                             (/RPLACA X NIL)
                             (RETURN (CONSCOUNT]
                6)
          (PROGN [IDIFFERENCE (CONSCOUNT)
                        (PROG (LISPXHIST)
                              (DECLARE (LOCALVARS . T))
                              (/RPLACA X NIL)
                              (RETURN (CONSCOUNT]
                7])
```

(.COND

```
  [LAMBDA (X Y Z)
    (LIST 1 (COND
                (X 1)
                ((NULL X)
                 2)
                (T 3))
          (COND
             ((COND
                 (X 1)
                 ((NULL X)
                  2)
                 (T 3))
              2)
             ((NOT (COND
                       (X 1)
                       ((NULL X)
                        2)
                       (T 3)))
              3)
             (T 4))
          (OR (COND
                 (X 1)
                 ((NULL X)
                  2)
                 (T 3))
              5)
          (AND (COND
                  (X 1)
                  ((NULL X)
                   2)
                  (T 3))
               6)
          (PROGN (COND
                    (X 1)
                    ((NULL X)
                     2)
                    (T 3))
                 7])
```

(.DELBIND

```
  [LAMBDA (X Y Z)
    (LIST 1 [PROG NIL
                  (RETURN (LIST 10 (COND
                                      (X (RETURN 11]
          (COND
             ([PROG NIL
                    (RETURN (LIST 10 (COND
                                       (X (RETURN 11]
              2)
             ([NOT (PROG NIL
                         (RETURN (LIST 10 (COND
                                            (X (RETURN 11]
              3)
             (T 4))
          (OR [PROG NIL
                    (RETURN (LIST 10 (COND
                                       (X (RETURN 11]
              5)
          (AND [PROG NIL
                     (RETURN (LIST 10 (COND
                                        (X (RETURN 11]
               6)
          (PROGN [PROG NIL
                       (RETURN (LIST 10 (COND
                                          (X (RETURN 11]
                 7])
```

```
)

(RPAQQ TESTC2
       [(TESTVALS (.SPEC '(NIL))
                '(1 0 2 0 6 7))
        (TESTVALS (.COND 3)
               '(1 1 2 1 6 7)
               (.COND)
               '(1 2 2 2 6 7))
        (TESTVALS (.DELBIND)
               '(1 (10 NIL)
                   2
                   (10 NIL)
                   6 7)
               (.DELBIND 10)
               '(1 11 2 11 6 7])

(DEFINEQ

(T1
  [LAMBDA (X)
    1])

(T0
  [LAMBDA (X)
    0])

(TT
  [LAMBDA (X)
    T])

(TNIL
  [LAMBDA (X)
    NIL])

(T-1
  [LAMBDA (X)
    -1])

(T2
  [LAMBDA (X)
    2])

(T12
  [LAMBDA (X)
    12])

(T377
  [LAMBDA (X)
    255])

(T400Q
  [LAMBDA (X)                                          ; Edited  8-Apr-88 13:22 by amd
    256])

(T-400
  [LAMBDA (X)
    -256])

(TSTR
  [LAMBDA (X)                                          ; Edited  8-Apr-88 13:22 by amd
    "FOO"])

(.NILARGS
  [LAMBDA (A B C D E F G H I J K L M N O)
    (DECLARE (LOCALVARS . T))
    (AND (OR A B C D E F G H I J K L M N O)
         (FAILTEST])

)

(RPAQQ TESTC
       [(TESTVALS (T1)
                1
                (T0)
```

```
              0
              (TT)
              T
              (TNIL)
              NIL
              (T-1)
              -1
              (T2)
              2
              (T12)
              12
              (T377)
              255
              (T400Q)
              256
              (T-400)
              -256
              (TSTR)
              "FOO")
       (PROG [(TEST '(.NILARGS]
             (RPTQ 1000 (.NILARGS])

(DEFINEQ

(.BIND0
  [LAMBDA (X)                                          ; Edited  8-Apr-88 13:22 by amd
    (PROG (Y Z W)
          (RETURN X])

(.BIND1
  [LAMBDA (A B)                                        ; Edited  8-Apr-88 13:23 by amd
                                                       ; lmm: 22-JUN-76 1 0
    (DECLARE (SPECVARS . T))
    (PROG ((C 1)
           (D 2))
          (RETURN (PROG (E F (G 3)
                             (H 4))
                        (RETURN (EVALVARS])

(.BIND2
  [LAMBDA (A B)                                        ; Edited  8-Apr-88 13:23 by amd
                                                       ; lmm: 22-JUN-76 1 0
    (DECLARE (LOCALVARS . T))
    (PROG ((C 1)
           (D 2))
          (RETURN (PROG (E F (G 3)
                             (H 4))
                        (RETURN (EVALVARS])

(.BIND3
  [LAMBDA (A B)                                        ; Edited  8-Apr-88 13:23 by amd
                                                       ; lmm: 24-JUN-76 4 57
    (DECLARE (SPECVARS . T))
    (LIST [PROG ((C 1)
                 (D 2))
                (RETURN (PROG (E F (G 3)
                                   (H 4))
                              (RETURN (EVALVARS]
          A B])

(.BIND4
  [LAMBDA (A B)                                        ; Edited  8-Apr-88 13:23 by amd
                                                       ; lmm: 24-JUN-76 4 57
    (DECLARE (SPECVARS . T))
    (LIST [PROG ((C 1)
                 (D 2))
                (RETURN (PROG (E F (G 3)
                                   (H 4))
                              (RETURN (EVALVARS]
          A B])

(.BINDASSOC
  [LAMBDA (V ALST VAR)                                 ; Edited  8-Apr-88 13:23 by amd
                                                       ; lmm: 24-JUN-76 8 42
    (PROG ((D (ASSOC V ALST)))
          (RETURN (EVAL VAR])

(.BIND5
  [LAMBDA (X)                                          ; Edited  8-Apr-88 13:23 by amd
                                                       ; lmm: 24-JUN-76 8 51
    (PROG ((D (CDR X)))
```

```
          (PROG ((LC (CDR D)))
                (DECLARE (LOCALVARS LC))
                ([LAMBDA (X)
                   (RETFROM '.BIND5 (EVAL 'X]
                 LC])
```

(.BINDPOP

```
  [LAMBDA (X Y)                                          ; Edited  8-Apr-88 13:24 by amd
                                                         ; lmm: 24-JUN-76 9 40
    (PROG1 Y
        (PROG ((K X))
              (CONS K K)
              (SETQ Y X)))])

)

(RPAQQ TESTBIND
       [(TESTVALS (.BIND0 173)
                  173
                  (.BIND1 1 2 3)
                  '(1 2 1 2 NIL NIL 3 4)
                  (.BIND2 1 2 3)
                  '(A.TOP B.TOP C.TOP D.TOP E.TOP F.TOP G.TOP H.TOP)
                  (.BIND3 1 2)
                  '((1 2 1 2 NIL NIL 3 4)
                    1 2)
                  (.BIND4 1 2 3 4)
                  '((1 2 1 2 NIL NIL 3 4)
                    1 2)
                  (.BINDASSOC 'A '((A . B))
                         'D)
                  '(A . B)
                  (.BINDPOP 23 73)
                  73
                  (.BIND5 '(A B C D E))
                  '(C D E])

(RPAQQ TESTAT
       ((TESTVALS (ARGTYPE 'LAM0)
                  0
                  (NARGS 'LAM0)
                  0
                  (ARGLIST 'LAM0)
                  NIL
                  (CALLS 'LAM0)
                  '(NIL NIL NIL NIL)
                  (FNTYP 'LAM0)
                  'CEXPR
                  (CCODEP 'LAM0)
                  T
                  (ARGTYPE 'LAM1)
                  0
                  (NARGS 'LAM1)
                  1
                  (ARGLIST 'LAM1)
                  '(X)
                  (CALLS 'LAM1)
                  '(NIL NIL NIL NIL)
                  (FNTYP 'LAM1)
                  'CEXPR
                  (CCODEP 'LAM1)
                  T
                  (ARGTYPE 'LAMA)
                  2
                  (NARGS 'LAMA)
                  1
                  (ARGLIST 'LAMA)
                  'U
                  (CALLS 'LAMA)
                  '(NIL NIL NIL NIL)
                  (FNTYP 'LAMA)
                  'CEXPR*
                  (CCODEP 'LAMA)
                  T
                  (ARGTYPE 'NLAML)
                  1
                  (NARGS 'NLAML)
                  1
                  (ARGLIST 'NLAML)
                  '(L)
                  (CALLS 'NLAML)
                  '(NIL NIL NIL NIL)
                  (FNTYP 'NLAML)
                  'CFEXPR
                  (CCODEP 'NLAML)
                  T
```

```
               (ARGTYPE 'NLAMA)
               3
               (NARGS 'NLAMA)
               1
               (CALLS 'NLAMA)
               '(NIL NIL NIL NIL)
               (FNTYP 'NLAMA)
               'CFEXPR*
               (CCODEP 'NLAMA)
               T)))

(DEFINEQ

(LAM0
  [LAMBDA NIL NIL])

(LAM1
  [LAMBDA (X)
    (DECLARE (LOCALVARS X))                          ; Edited 8-Apr-88 13:24 by amd
    X])

(LAMA
  [LAMBDA L
    (DECLARE (SPECVARS L))                           ; Edited 8-Apr-88 13:24 by amd
    L])

(NLAML
  [NLAMBDA (L)
    (DECLARE (LOCALVARS L))                          ; Edited 8-Apr-88 13:24 by amd
    L])

(NLAMA
  [NLAMBDA L
    (DECLARE (LOCALVARS L))                          ; Edited 8-Apr-88 13:25 by amd
    L])

)

(DEFINEQ

(.SELECTQ
  [LAMBDA (A B C)                                    ; Edited 8-Apr-88 13:25 by amd
    (LIST 3 (PROGN (SELECTQ A
                           (1 (ADD1VAR C))
                           ((2 3 4)
                               (SUB1VAR C))
                           (5 (SETQ C (CDR C)))
                           NIL)
                   (SELECTQ B
                           (1 (ADD1VAR C))
                           ((2 3 4)
                               (SUB1VAR C))
                           (5 (SETQ C (CDR C)))
                           NIL))
          (COND
             ((SELECTQ C
                   (NIL T)
                   (0 NIL)
                   (3 (SMALLP B))
                   A)
              22])

(.SUBFNS
  [LAMBDA NIL                                        ; Edited 8-Apr-88 13:25 by amd
    (LIST (PROG1 'GOOD
              [SETQ FREE1 (FUNCTION (LAMBDA (X)
                                        (CAR X]
              [SETQ FREE2 (FUNCTION (LAMBDA N N]
              [SETQ FREE3 (FUNCTION (NLAMBDA L L])
          (APPLY* FREE1 '(A)
                 '(B))
          (APPLY* FREE2 1 2 3 4 5)
          (APPLY* FREE3 1 2 3 4 5)
          (SUBSET '((NIL)
                    (3))
                 FREE1])

(.MISC
  [LAMBDA (B C A)                                    ; Edited 8-Apr-88 13:25 by amd
    (LIST 1 (PROG ((A NIL))
                  (RETURN (PROGN B C A))
```

```
                (FOO BAZ WHAMMY)
                (PROG (X)
                      (HELP)))
          3])
```

(.FORTEST

```
  [LAMBDA (X)                                                    ; Edited 8-Apr-88 13:25 by amd
    (for X on (to X collect (to X collect X)) when (SOME X (FUNCTION CDDR)) collect (CONS X (LENGTH X])
```

(.BIGCOND

```
  [LAMBDA (X)                                                    ; Edited 8-Apr-88 13:25 by amd
    (COND
       ((LISTP X)
        (LIST X))
       ((ARRAYP X)
        (ELT X 1))
       ((FIXP X)
        (ITIMES (ITIMES 60 24 365)
               X])
```

(.RECORDTEST

```
  [LAMBDA (ARG)                                                  ; Edited 8-Apr-88 13:26 by amd
                                                                 (* DECLARATIONS%: (RECORD A
                                                                 (B . C)))
    (PROG [(ZZ (create A
                       C _ (RPLACA (CONS)
                                   1)
                       B _ (RPLACA (CONS)
                                   2]
          (COND
             (ARG (replace B of ZZ with 17)))
          [RPLACD (fetch C of ZZ)
                  (create A
                         C _ (LIST 4)
                         B _ (COND
                                ((ZEROP ARG)
                                 (GO HOME))
                                ((EQ ARG 3)
                                 (RETURN ZZ))
                                (T -2]
      HOME
          (RETURN ZZ])
```

(.PROGRETURN

```
  [LAMBDA (X)                                                    ; Edited 8-Apr-88 13:26 by amd
    (PROG NIL
          (SETQ X (LIST (LIST 1)
                        (LIST 2)))
          (RPLACD (ASSOC 1 X)
                  T)
          (RETURN (SOME X (FUNCTION LISTP])
```

(.ALWAYSFALSE

```
  [LAMBDA (A B)                                                  ; Edited 8-Apr-88 13:26 by amd
    (LIST 1 (COND
                ((COND
                    (A NIL)
                    (B NIL))
                 T)
                (T NIL))
          3])
```

(.ALWAYSTRUE

```
  [LAMBDA (A B)                                                  ; Edited 8-Apr-88 13:26 by amd
    (COND
       ((COND
           (A T)
           (B T)
           (T T))
        T)
       (T NIL])
```

(.EQ1

```
  [LAMBDA (X Y)
    (DECLARE (LOCALVARS X Y))                                    ; Edited 8-Apr-88 13:27 by amd
    (COND
       ((EQ X Y)
        T)
       [(NLISTP X)
        (COND
```

```
           ((OR (NUMBERP X)
                (STACKP X))
            (EQP X Y))
           ((STRINGP X)
            (STREQUAL X Y]
       ((LISTP Y)
        (AND (.EQ1 (CAR X)
                   (CAR Y))
             (.EQ1 (CDR X)
                   (CDR Y])
```

(**.EQ2**

```
  [LAMBDA (X Y)
    (DECLARE (LOCALVARS X Y))                          ; Edited 8-Apr-88 13:27 by amd
    (COND
       [(NEQ X Y)
        (COND
           [(LISTP X)
            (AND (LISTP Y)
                 (.EQ2 (CAR X)
                       (CAR Y))
                 (.EQ2 (CDR X)
                       (CDR Y]
           [(NOT (OR (NUMBERP X)
                     (STACKP X)))
            (COND
               ((STRINGP X)
                (STREQUAL X Y]
           (T (EQP X Y]
       (T T])
```

(**.EQ3**

```
  [LAMBDA (X Y)
    (DECLARE (LOCALVARS X Y))                          ; Edited 8-Apr-88 13:27 by amd
    (COND
       ((EQ X Y)
        T)
       ((LISTP X)
        (COND
           [(LISTP Y)
            (AND (.EQ3 (CAR Y)
                       (CAR X))
                 (.EQ3 (CDR X)
                       (CDR Y]
           (T NIL)))
       ((OR (NUMBERP X)
            (STACKP X))
        (EQP X Y))
       ((STRINGP X)
        (STREQUAL X Y))
       (T NIL])

)

(RPAQQ TEST3
       ((TESTVALS (.SELECTQ 1 1 2)
                  '(3 4 22)
                  (.SELECTQ 1 3 3)
                  '(3 3 22)
                  (.SELECTQ 1 3 0)
                  '(3 0 NIL)
                  (.SELECTQ 2 2 2)
                  '(3 0 NIL)
                  (.SELECTQ 5 5 '(NIL NIL . 3))
                  '(3 3 22)
                  (.SELECTQ)
                  '(3 NIL 22)
                  (.SUBFNS)
                  '(GOOD A 5 (1 2 3 4 5)
                         ((3)))
                  (.MISC)
                  '(1 NIL 3)
                  (.FORTEST 3)
                  '[(((3 3 3)
                      (3 3 3)
                      (3 3 3)) . 3)
                    (((3 3 3)
                      (3 3 3)) . 2)
                    (((3 3 3)) . 1]
                  (.FORTEST 2)
                  NIL)))

(DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVARS

(ADDTOVAR NLAMA NLAMA TESTIFSAME TESTVALS)
```

```
(ADDTOVAR NLAML NLAML !ADD1VAR .NLSETQ TESTPAIRS)

(ADDTOVAR LAMA LAMA !APPEND .LIST .SUM .TESTARG)
)

(PUTPROPS COMPTEST COPYRIGHT ("Venue & Xerox Corporation" 1984 1985 1988 1990))
```

FUNCTION INDEX

!ADD1VAR 14
!AND 13
!APPEND 14
!ASSOC 15
!ATTACH 15
!CHANGEPROP ..15
!COPY 15
!DEFLIST 15
!DREMOVE 15
!DREVERSE 16
!DSUBST 16
!EVERY 16
!GET 18
!GETLIS 18
!GETP 16
!INTERSECTION 17
!LAST 17
!LASTN 17
!LDIFF 17
!LENGTH 17
!LISTGET 17
!LSUBST 18
!MAP 18
!MEMB 18
!NTH 18
!OR 13
..FRPLNODE2 ..13
..LAM0 27
..LAM1 27
..NCONC 12
..NCONC3 12
.ADD1 11
.ALWAYSFALSE .34
.ALWAYSTRUE ..34
.AND 12
.ANDARRAYP 7
.ANDATOM 7
.ANDFLOATP 7
.ANDLISTP 6
.ANDNATOM 7
.ANDNLISTP 7
.APPEND0 19
.APPEND1 19
.APPEND2 19
.APPEND3 19
.APPEND4 19
.ARRAYP 5
.ASSOC 24
.ATOM 6
.ATTACH 19
.BIGCOND 34
.BIND0 31
.BIND1 31
.BIND2 31
.BIND3 31
.BIND4 31
.BIND5 31
.BINDASSOC ...31
.BINDPOP 32
.CAAAAR 27
.CAAR 26
.CADR 26
.CAR 26
.CDAR 26
.CDDR 27
.CDR 26
.COLLCT 18
.COND 29
.CONDRET 8
.CONDRET2 8
.CONS 25
.DELBIND 29
.ENDCOLLCT ...19
.EQ 6
.EQ1 34
.EQ2 35
.EQ3 35
.EQMEMB 13
.EQUAL 10
.EVALV1 9
.EVALV2 9
.FDIFFERENCE .22
.FGETD 13
.FIX 9
.FIXP 5
.FLOATP 4
.FMEMB 24
.FORTEST 34
.FPLUS 22
.FQUOTIENT ...22
.FRPLACA 24
.FRPLACD 24
.FRPLNODE 12
.FRPLNODE2 ...12
.FTIMES 22
.GET 24
.GETHASH 24
.GETP 24
.GETPROP 24
.GETPROPLIST .13
.GETPROPLST ...4
.IDIFFERENCE .11
.IEQP 6
.IGREATERP 6
.ILESSP 6
.IPLUS 10
.IQUOTIENT ...10
.IREMAINDER ..10
.ITIMES 10
.IVAR 23
.LAM0 27
.LAM1 27
.LAST 24
.LENGTH 24
.LIST 9
.LISTP 4
.LITATOM 4
.LLSH 11
.LOGAND 11
.LOGOR 11
.LOGXOR 11
.LRSH 11
.LSH 11
.MAP 21
.MAPC 21
.MAPCEFF 22
.MAPCGETP 19
.MISC 33
.MKLIST 13
.NARRAYP 5
.NATOM 6
.NCONC 12
.NCONC1 13
.NCONC3 12
.NEQ 6
.NFIXP 5
.NFLOATP 5
.NIEQP 6
.NIGREATERP ...6
.NILARGS 30
.NILESSP 6
.NLAML 27
.NLISTP 4
.NLITATOM 5
.NLSETQ 9
.NNULL 6
.NNUMBERP 5
.NSMALLP 5
.NSTACKP 5
.NSTRINGP 6
.NT1 11
.NT2 11
.NT3 11
.NULL 6
.NUMBERP 5
.NZEROP 5
.OR 12
.ORARRAYP 7
.ORATOM 6
.ORFLOATP 7
.ORLISTP 6
.ORNULL 7
.ORZEROP 7
.PROGRETURN ..34
.PROGS 28
.RECORDTEST ..34
.RPLACA 24
.RPLACD 24
.RSH 11
.SELECTQ 33
.SET 9
.SETPROPLIST .13
.SETX 10
.SMALLP 5
.SPEC 28
.STACKP 5
.STRINGP 5
.SUB1 11
.SUBFNS 33
.SUM 9
.TESTARG 9
.ZEROP 5
APPEND2 15
CNTDWN 23
COLLCT 19
DREV 16
ENDCOLLCT 19
EVALVARS 3
FAILTEST 2
FN1 23
FN2 23
FN3 23
FVAR 4
GETLEAVES 8
IVAR 3
IVAR3 3
IVARX 4
JUMPAROUND ...23
LAM0 33
LAM1 33
LAM1LOC 27
LAMA 33
LIST0 25
LIST1 25
LIST10 26
LIST11 26
LIST12 26
LIST13 26
LIST2 25
LIST3 25
LIST4 25
LIST5 25
LIST6 25
LIST7 25
LIST8 25
LIST9 26
MAKEFORM 1
MYAPPEND1 19
MYAPPEND2 19
NLAMA 33
NLAML 33
NONLOCALGO ...23
PVAR 3
SELECTTEST ...12
T-1 30
T-400 30
T0 30
T1 30
T12 30
T2 30
T377 30
T400Q 30
TEST1 2
TESTER 2
TESTIFSAME 2
TESTPAIRS 2
TESTRESUME 8
TESTVALS 2
TNIL 30
TRY 2
TRYTEST 3
TSTR 30
TT 30
VARSWAP 3
VARSWAP2 3
VARSWAP3 3
VARSWAP4 3

VARIABLE INDEX

TEST3 ....35 TESTC ....30 TESTCAR ..27 TESTFN ...24 TESTLINK .22 TESTMORE .13 TESTRET ...9 TESTVAR ...4
TESTAT ...32 TESTC2 ...30 TESTCONS .26 TESTFNX ..23 TESTMAP ..22 TESTNUM ..11 TESTS 3
TESTBIND .32 TESTCALL .27 TESTEDIT .25 TESTJUMP .23 TESTMISC .19 TESTOP ...10 TESTTYPE ..7