File created: 29-Nov-2021 14:05:45 **{DSK}<Users>kaplan>Local>medley3.5>my-medley>sources>COMPARE.;3**

changes to: (FNS COMPARE1)

previous date: 5-Nov-2021 20:53:09 {DSK}<Users>kaplan>Local>medley3.5>my-medley>sources>COMPARE.;2

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

```
;;
;; Copyright (c) 1987, 1990 by Venue & Xerox Corporation.

(RPAQQ COMPARECOMS
       ((FNS COMPARELST COMPARE1 COMPAREPRINT COMPAREPRINT1 COMPARELISTS COMPAREPRINTN COMPARENCHARS COMPAREFAIL
             COMPAREMAX COUNTDOWN)
        (ADDVARS (COMPARETRANSFORMS))
        (INITVARS (*COMPARE-LEVEL* 1)
               (*COMPARE-LENGTH* 2))
        (SPECVARS *COMPARE-LEVEL* *COMPARE-LENGTH* DIFFERENCES LOOSEMATCH)
        (GLOBALVARS COMPARETRANSFORMS COMMENTFLG)))

(DEFINEQ
```

(COMPARELST

```
  [LAMBDA (X Y LOOSEMATCH)                                  ; Edited 20-Jan-87 12:38 by bvm:
    (DECLARE (SPECVARS LOOSEMATCH))
    [COND
       ((EQ LOOSEMATCH -1)
        (SETQ LOOSEMATCH (COMPAREMAX X Y]
    (COMPARE1 X Y])
```

(COMPARE1

```
  [LAMBDA (X Y)                                             ; Edited 29-Nov-2021 13:49 by rmk:
                                                            ; (* lmm "29-AUG-78 18:35")
    ;; returns T if X and Y are similar;  if LOOSEMATCH then sets DIFFERENCES to changes
    (AND [OR (EQ X Y)
             (COND
                [(LISTP X)
                 (COND
                    [(LISTP Y)
                     (OR (AND (EQ (CAR X)
                                  COMMENTFLG)
                              (EQ (CAR Y)
                                  COMMENTFLG))
                         (PROG NIL
                           LP  (RETURN (COND
                                         ((NLISTP X)
                                          (OR (EQUALALL X Y)
                                              (COMPAREFAIL X Y)))
                                         ((NLISTP Y)
                                          (COMPAREFAIL X Y))
                                         ((NOT (COMPARE1 (CAR X)
                                                         (CAR Y)))
                                          NIL)
                                         (T (SETQ X (CDR X))
                                            (SETQ Y (CDR Y))
                                            (GO LP]
                         (T (COMPAREFAIL X Y]
                    (T (OR (EQUALALL X Y)
                           (COMPAREFAIL X Y]
         (OR LOOSEMATCH T])
```

(COMPAREPRINT

```
  [LAMBDA (X Y STREAM)                                      ; Edited 20-Jan-87 12:20 by bvm:
    (PROG ((*PRINT-LEVEL* *COMPARE-LEVEL*)
           (*PRINT-LENGTH* *COMPARE-LENGTH*)
           (PLVLFILEFLG T)
           FIN)
          (COND
             ((EQUAL X Y)
              (RETURN NIL)))
          (COND
             ((OR (NLISTP X)
                  (NLISTP Y))
              (PRINT X STREAM)
              (PRINT Y STREAM)
              (GO FIN)))
          (PRIN1 '%( STREAM)                                ; Print list X by comparison with list Y
          (COMPAREPRINT1 X Y STREAM)
          (PRIN1 '%) STREAM)
```

```
          (TERPRI STREAM)
          (PRIN1 '%( STREAM)                                    ; Do same for other list
          (COMPAREPRINT1 Y X STREAM)
          (PRIN1 '%) STREAM)
          (TERPRI STREAM)
      FIN (RETURN T])
```

(COMPAREPRINT1

```
  [LAMBDA (A B STREAM)                                          ; Edited 20-Jan-87 12:28 by bvm:
```

;;; [JDS: Added STREAM argument to direct output.]

```
    (PROG ((N 0)
           X Y SPACE DOTFLAG L1 TAILX TAILY K)
          (SETQ TAILX A)
          (SETQ TAILY B)
      L1  [COND
             (DOTFLAG (SETQ X TAILX)
                    (SETQ Y TAILY))
             (T (SETQ X (CAR TAILX))
                (SETQ Y (CAR TAILY]
          [COND
             ((EQ (SETQ K (COMPAREMAX X Y))
                  (SETQ K (COMPARELST X Y K)))                  ; If two sublists are the same just type '&'
              (COND
                 ((AND (NOT SPACE)
                       (LITATOM X)
                       (EQ N 0))
                  (CL:PRIN1 X STREAM)
                  (GO NX1))
                 (T (ADD1VAR N)
                    (GO NX]
          (COMPAREPRINTN N SPACE T STREAM)
          (SETQ N 0)
          (COND
             ((OR (NLISTP X)
                  (NLISTP Y)))
             [(EQ (CAR X)
                  COMMENTFLG)
              (PRIN1 **COMMENT**FLG STREAM)
              (COND
                 ((NEQ (CAR Y)
                       COMMENTFLG)
                  (SETQ TAILX (CDR TAILX))
                  (GO L1]
             ((EQ (CAR Y)
                  COMMENTFLG)
              (SPACES (NCHARS **COMMENT**FLG)
                      STREAM)
              (SETQ TAILY (CDR TAILY))
              (GO L1)))
          [COND
             ((AND (NULL K)
                   (NULL DOTFLAG))
              (COND
                 ((AND (LISTP TAILX)
                       (LISTP (CDR TAILX))
                       (COMPARELST (CADR TAILX)
                                   Y -1))                       ; Next X same as this Y, so just have an inserted item
                  (CL:PRIN1 X STREAM)
                  (SETQ TAILX (CDR TAILX))
                  (GO L1))
                 ((AND (LISTP TAILY)
                       (LISTP (CDR TAILY))
                       (COMPARELST (CADR TAILY)
                                   X -1))                       ; Next Y same as this X, so leave space corresponding to the
                                                                ; inserted item
                  (SPACES (COND
                             ((NLISTP Y)
                              (NCHARS Y T))
                             (T                                 ; List would be printed at print level 1, so count carefully
                                (COMPARENCHARS Y)))
                          STREAM)
                  (SETQ TAILY (CDR TAILY))
                  (GO L1]
          (COND
             ((OR (NLISTP X)
                  (NLISTP Y))                                   ; If they are unequal and one is not a list let PRIN2 type out
                                                                ; something (atom or list)
              (CL:PRIN1 X STREAM))
             (T (PRIN1 '%( STREAM)                              ; Otherwise print '()' and subanalyze
                (COMPAREPRINT1 X Y STREAM)
                (PRIN1 '%) STREAM)))
      NX1 (SETQ SPACE T)
      NX  (COND
             ((OR DOTFLAG (NLISTP TAILX)
```

```
                (NOT (CDR TAILX)))                          ; X list ran out
            (COMPAREPRINTN N SPACE NIL STREAM))
           (T (SETQ DOTFLAG (NLISTP (CDR TAILX)))
              (COND
                 ((CDR (LISTP TAILY))
                  (SETQ TAILX (CDR TAILX))
                  (SETQ TAILY (CDR TAILY))
                  (GO L1)))
              (COMPAREPRINTN N SPACE NIL STREAM)
              (COND
                 (DOTFLAG (PRIN1 '" . " STREAM)
                        (CL:PRIN1 (CDR TAILX)
                                  STREAM))
                 (T                                         ; (CDR TAILX) is a list
                    (SPACES 1 STREAM)
                    (CL:PRIN1 (CADR TAILX)
                              STREAM)
                    (AND (CDDR TAILX)
                         (PRIN1 '" --" STREAM])
```

(COMPARELISTS

```
[LAMBDA (X Y STREAM)                                        ; Edited 20-Jan-87 12:39 by bvm:
```

;; functionally equivalent to CPLISTS . Prints differences on STREAM.

```
(SETQ STREAM (GETSTREAM STREAM 'OUTPUT))
(PROG (DIFFERENCES)                                         ; DIFFERENCES used by COMPAREFAIL
      (DECLARE (SPECVARS DIFFERENCES))
      (COND
         ((NOT (COMPARELST X Y T))                          ; lists are different enough to require play by play display
          (COMPAREPRINT X Y STREAM))
         [DIFFERENCES                                       ; x and y are different, but only by substitution.  Each element of
                                                            ; differences is a dotted pair
                (for TAIL on DIFFERENCES do (LET ((PAIR (CAR TAIL)))
                                                 (CL:FORMAT STREAM "~S -> ~S" (CAR PAIR)
                                                        (CDR PAIR))
                                                 (if (CDR TAIL)
                                                     then (PRIN1 ", " STREAM]
         (T (PRIN1 'SAME STREAM)))
      (TERPRI STREAM])
```

(COMPAREPRINTN

```
[LAMBDA (N SPACE FLG STREAM)                                ; Edited  5-Nov-2021 20:53 by rmk:
                                                            ; Edited 29-Dec-86 11:56 by jds
```

;; RMK: Added STREAM to POSITION and LINELENGTH

```
[COND
   ((NEQ N 0)
    (COND
       (SPACE (SPACES 1 STREAM))
       (T (SETQ SPACE T)))
    (SELECTQ N
         (1 (PRIN1 '& STREAM))
         (PROGN (COND
                   ((NOT (ILESSP (IPLUS (POSITION STREAM)
                                        7)
                                 (LINELENGTH NIL STREAM)))
                    (TERPRI STREAM)))
                (PRIN1 '- STREAM)
                (PRIN2 N STREAM)
                (PRIN1 '- STREAM]
(AND FLG SPACE (SPACES 1 STREAM])
```

(COMPARENCHARS

```
[LAMBDA (X)                                                 ; Edited 20-Jan-87 12:26 by bvm:
```

;; Count the number of characters that would be printed at the current print depth

```
(LET [(COMPARECNT 0)
      (*PRINT-ESCAPE* T)
      (*READTABLE* (\DTEST *READTABLE* 'READTABLEP]
     (DECLARE (SPECVARS *READTABLE* *PRINT-ESCAPE*))
     (\MAPPNAME.INTERNAL [FUNCTION (LAMBDA (S C)
                                   (ADD COMPARECNT 1]
            X)
     COMPARECNT])
```

(COMPAREFAIL

```
[LAMBDA (X Y)                                               ; Edited 13-Jan-87 14:29 by bvm:
```

;; X and Y are different. Return non-NIL if we are willing to believe that X and Y really are the same for purposes of not going thru
;; COMPAREPRINT. DIFFERENCES is a list associating occurrences of X with a value of Y; if all such occurrences are the same, then
;; COMPARELST will just print a series of transformations X -> Y.

```
(COND
   [(SOME COMPARETRANSFORMS (FUNCTION (LAMBDA (FN)
```

```
                                          (CL:FUNCALL FN X Y]
       ((NULL LOOSEMATCH)                                          ; exact match demanded
        NIL)
       ((NUMBERP LOOSEMATCH)
        (IGREATERP [SETQ LOOSEMATCH (COUNTDOWN Y (COUNTDOWN X (SUB1 LOOSEMATCH]
              0))
       ([AND (NLISTP X)
             (OR (NLISTP Y)
                 (EVERY Y (FUNCTION NLISTP]
        (LET ((OLD (FASSOC X DIFFERENCES)))
             (if OLD
                 then (EQUAL Y (CDR OLD))
               else (SETQ DIFFERENCES (NCONC1 DIFFERENCES (CONS X Y])
```

(COMPAREMAX

```
  [LAMBDA (X Y)                                                  (* lmm "30-AUG-78 02:19")
    (IQUOTIENT (IDIFFERENCE 65 (IPLUS (COUNTDOWN X 30)
                                      (COUNTDOWN Y 30)))
          5])
```

(COUNTDOWN

```
  [LAMBDA (X N)                                                  (* lmm "30-AUG-78 02:37")
    (COND
       ((OR (NLISTP X)
            (NOT (IGREATERP N 0)))
        N)
       (T (COUNTDOWN (CDR X)
                     (COUNTDOWN (CAR X)
                            (SUB1 N])

)

(ADDTOVAR COMPARETRANSFORMS )

(RPAQ? *COMPARE-LEVEL* 1)

(RPAQ? *COMPARE-LENGTH* 2)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(SPECVARS *COMPARE-LEVEL* *COMPARE-LENGTH* DIFFERENCES LOOSEMATCH)
)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS COMPARETRANSFORMS COMMENTFLG)
)

(PUTPROPS COMPARE COPYRIGHT ("Venue & Xerox Corporation" 1987 1990))
```

FUNCTION INDEX

VARIABLE INDEX