*File created:* 22-Mar-95 18:19:18 **{DSK}<lispcore>library>new>PCTREE.;1**

*changes to:* (FNS \INSERTTREE \DELETETREE \SPLITTREE \TEDIT.UPDATETREE)

*previous date:* 7-Oct-94 17:44:31 {DSK}<lispcore>library>PCTREE.;5

*Read Table:* INTERLISP

*Package:* INTERLISP

*Format:* XCCS

;;


```
(RPAQQ PCTREECOMS
       [;; Balanced tree PIECE TABLE supporting functions
        (FILES TEDITDCL)
        (DECLARE%: EVAL@COMPILE DONTCOPY
               ;; \WORDSINBTREEMAIN = # of words in the child-pointers & offsets section of the node -- everything before SPARE5 (the
               ;; overflow place).
               ;; \BTREEMAXCOUNT = number of children in a full node = maximum value for a node's COUNT field.
               ;; \BTREELASTREALOFFSET = offset of last real space for a child entry in the node ( = \WORDSINBTREEMAIN - 4)
               (CONSTANTS (\BTREEMAXENTRIES 8)
                      (\BTREEMAXCOUNT 8)
                      (\BTREEWORDSPERENTRY 4)
                      (\WORDSINBTREEMAIN (UNFOLD \BTREEMAXENTRIES 4))
                      (\BTREELASTREALOFFSET (UNFOLD (SUB1 \BTREEMAXENTRIES)
                                                    4))
                      (\BTREETOPHALFOFFSET (UNFOLD (LRSH \BTREEMAXENTRIES 1)
                                                   4)))
               (FILES (LOADCOMP)
                      TEDITDECLS))
        (FNS UPDATEPCNODES FINDPCNODE \FIRSTNODE \DELETETREE \INSERTTREE \LASTNODE \MATCHPCS \SPLITTREE
             \TEDIT.UPDATETREE \TEDIT.PIECE-CHNO \TEDIT.SET-TOTLEN)
        (FNS DISPTREE TREEGRAPHNODE)
        (RECORDS BTREENODE)
        (DECLARE%: DONTEVAL@LOAD DOEVAL@COMPILE DONTCOPY COMPILERVARS (ADDVARS (NLAMA)
                                                                                (NLAML)
                                                                                (LAMA])
```

;; Balanced tree PIECE TABLE supporting functions

```
(FILESLOAD TEDITDCL)

(DECLARE%: EVAL@COMPILE DONTCOPY

(DECLARE%: EVAL@COMPILE

(RPAQQ \BTREEMAXENTRIES 8)

(RPAQQ \BTREEMAXCOUNT 8)

(RPAQQ \BTREEWORDSPERENTRY 4)

(RPAQ \WORDSINBTREEMAIN (UNFOLD \BTREEMAXENTRIES 4))

(RPAQ \BTREELASTREALOFFSET (UNFOLD (SUB1 \BTREEMAXENTRIES)
                                   4))

(RPAQ \BTREETOPHALFOFFSET (UNFOLD (LRSH \BTREEMAXENTRIES 1)
                                  4))

(CONSTANTS (\BTREEMAXENTRIES 8)
       (\BTREEMAXCOUNT 8)
       (\BTREEWORDSPERENTRY 4)
       (\WORDSINBTREEMAIN (UNFOLD \BTREEMAXENTRIES 4))
       (\BTREELASTREALOFFSET (UNFOLD (SUB1 \BTREEMAXENTRIES)
                                     4))
       (\BTREETOPHALFOFFSET (UNFOLD (LRSH \BTREEMAXENTRIES 1)
                                    4)))
)

(FILESLOAD (LOADCOMP)
       TEDITDECLS)
)

(DEFINEQ
```

## (UPDATEPCNODES

```
  [LAMBDA (PC DELTA)                                    ; Edited 21-Apr-93 16:09 by jds
    ;; ADD DELTA TO CHNUM IN NEXTALL NODES OF TOPNODE.
```

```
    (LET ((UPWARD (fetch (PIECE PTREENODE) of PC)))
         (while UPWARD do (for I from 0 by 4 as ITEM from 1 to (fetch (BTREENODE COUNT) of UPWARD)
                             when (EQ PC (\GETBASEPTR UPWARD I))
                             do [\PUTBASEFIXP UPWARD (IPLUS I 2)
                                      (IPLUS DELTA (\GETBASEFIXP UPWARD (IPLUS I 2]
                                (add (fetch (BTREENODE TOTLEN) of UPWARD)
                                     DELTA)
                                (SETQ PC UPWARD)
                                (SETQ UPWARD (fetch (BTREENODE UPWARD) of PC))
                                (RETURN)
                             finally (HELP "Piece not in its TREENODE"])
```

## (**FINDPCNODE**

```
  [LAMBDA (PC PCTB)                                          ; Edited 13-Apr-93 15:00 by jds
    ;; Given a piece and the pctb it's in, return pcnode
    (fetch (PIECE PTREENODE) of PC])
```

## (**\FIRSTNODE**

```
  [LAMBDA (TREE)                                             ; Edited 14-Apr-93 02:06 by jds
    (LET ((COUNT (fetch (BTREENODE COUNT) of TREE))
          CHILD)
         (SETQ CHILD (\GETBASEPTR TREE 0))
         (COND
            ((type? BTREENODE CHILD)
             (\FIRSTNODE CHILD))
            (T TREE])
```

## (**\DELETETREE**

```
  [LAMBDA (OLD PCNODE)                                       ; Edited 21-Mar-95 15:29 by sybalsky:mv:envos
    ;; Removes OLD from PCNODE.  OLD is either a piece or tree node.
    (UNINTERRUPTABLY
        (LET* ((OLDLEN (ffetch (BTREENODE TOTLEN) of PCNODE))
               NEWLEN INCHNO AFTERFLG NODE-COUNT ITEM# BB)
              ;; NEW CODE
              (SETQ NODE-COUNT (fetch (BTREENODE COUNT) of PCNODE))
              ;; Find OLD, .
              (for old ITEM# from 0 to (LLSH (SUB1 NODE-COUNT)
                                             2)
                 by 4 when (EQ OLD (\GETBASEPTR PCNODE ITEM#)) do (RETURN) finally (HELP "Piece/node not in
                                                                                          PCNODE"))
              ;; Update the previous piece's length, if appropriate:
              (SETQ BB (\ADDBASE PCNODE ITEM#))
              (\RPLPTR BB 0 NIL)
              [for I from 0 to (IDIFFERENCE \BTREELASTREALOFFSET ITEM#) by 4
                 do (\PUTBASEPTR BB I (\GETBASEPTR BB (IPLUS I 4)))
                    (\PUTBASEFIXP BB (IPLUS I 2)
                           (\GETBASEFIXP BB (IPLUS I 6]
              (\PUTBASEPTR PCNODE \BTREELASTREALOFFSET NIL)      ; Because it's been copied, clear the old value before the
                                                                ; refcnt-er gets to it.
              ;;  If adding this piece EMPTIES the tree node, DELETE it.
              ;; FIXMI -- This should coalesce adjacent nodes that are too empty!
              [COND
                 ((IEQP NODE-COUNT 1)
                  (\DELETETREE PCNODE (fetch (BTREENODE UPWARD) of PCNODE)))
                 (T                                             ; No split, so update upper nodes with delta-length.
                    [SETQ NEWLEN (replace (BTREENODE TOTLEN) of PCNODE
                                    with (for I from 2 to NODE-COUNT as ITEM# from 2 by 4
                                            sum (\GETBASEFIXP PCNODE ITEM#]
                    (replace (BTREENODE COUNT) of PCNODE with (SUB1 NODE-COUNT))
                    (\TEDIT.UPDATETREE PCNODE (IDIFFERENCE NEWLEN OLDLEN]
              ;; END NEW CODE
              1))])
```

## (**\INSERTTREE**

```
  [LAMBDA (NEW OLD PCNODE NEW-PREVLEN NEW-OLDLEN PREV)       ; Edited 22-Mar-95 15:37 by sybalsky:mv:envos
    ;; inserts NEW in front of OLD in PCNODE.  NEW/OLD are either pieces or tree nodes.
    ;; If NEWE-PREVLEN is non-NIL, it's a DELTA for updating parents of THE PIECE BEFORE OLD.  This is used by \SPLITPIECE to pass down the
    ;; new shortened length for the original piece.
    (UNINTERRUPTABLY
        (LET* ((OLDLEN (ffetch (BTREENODE TOTLEN) of PCNODE))
               NEWLEN INCHNO AFTERFLG NODE-COUNT ITEM# BB)
              ;; NEW CODE
              (SETQ NODE-COUNT (fetch (BTREENODE COUNT) of PCNODE))
```

```
        ;; Find OLD, and insert the NEW piece (and length) in front of it.
        [for old ITEM# from 0 to (LLSH (SUB1 NODE-COUNT)
                                       2)
           by 4 when (EQ OLD (\GETBASEPTR PCNODE ITEM#)) do (RETURN)
           finally (COND
                     (OLD (HELP "Old piece not in this PCNODE."))
                     (T                                          ; INSERTING FIRST PIECE
                        (SETQ ITEM# 0]
        (OR NEW (HELP "Inserting empty item"))
        ;; Update the previous piece's length, if appropriate:
        [AND NEW-PREVLEN (COND
                           ((ZEROP ITEM#)
                            ;; The hard way -- the previous piece is in a prior btree node, so we have to go there to update it.
                            (LET* ((NODE (fetch (PIECE PTREENODE) of PREV)))
                                  (UPDATEPCNODES PREV NEW-PREVLEN)))
                           (T ;; Easy way -- it's in this node.  Update it in place.
                              (\PUTBASEFIXP PCNODE (IDIFFERENCE ITEM# 2)
                                     (IPLUS NEW-PREVLEN (\GETBASEFIXP PCNODE (IDIFFERENCE ITEM# 2]
        (COND
           (NEW-OLDLEN (\PUTBASEFIXP PCNODE (IPLUS ITEM# 2)
                                     NEW-OLDLEN)))
        (SETQ BB (\ADDBASE PCNODE ITEM#))
        (\RPLPTR PCNODE \WORDSINBTREEMAIN NIL)                   ; Clean out the slot that's about to be copied over.
        (\BLT (\ADDBASE BB 4)
              BB
              (IDIFFERENCE \WORDSINBTREEMAIN ITEM#))
        (\PUTBASEPTR PCNODE ITEM# NIL)                           ; Because it's been copied, clear the old value before the
                                                                 ; refcnt-er gets to it.
        (\RPLPTR PCNODE ITEM# NEW)
        (COND
           ((type? PIECE NEW)
            (\PUTBASEFIXP PCNODE (IPLUS ITEM# 2)
                   (fetch (PIECE PLEN) of NEW))
            (replace (PIECE PTREENODE) of NEW with PCNODE))
           ((type? BTREENODE NEW)                                ; Inserting a NODE
            (\PUTBASEFIXP PCNODE (IPLUS ITEM# 2)
                   (fetch (BTREENODE TOTLEN) of NEW))
            (replace (BTREENODE UPWARD) of NEW with PCNODE))
           (T (\ILLEGAL.ARG NEW)))
        [SETQ NEWLEN (replace (BTREENODE TOTLEN) of PCNODE
                        with (for I from 0 to NODE-COUNT as ITEM# from 2 by 4 sum (\GETBASEFIXP PCNODE ITEM#
                                                                                                        ]
        ;; If adding this piece overflows the tree node, split it.
        [COND
           ((IEQP NODE-COUNT \BTREEMAXCOUNT)                     ; Tree node is full, so have to split.
            (\SPLITTREE PCNODE OLD NEW))
           (T                                                    ; No split, so update upper nodes with delta-length.
              (replace (BTREENODE COUNT) of PCNODE with (ADD1 NODE-COUNT))
              (\TEDIT.UPDATETREE PCNODE (IDIFFERENCE NEWLEN OLDLEN]
        ;; END NEW CODE
        1))])
```

## (\LASTNODE

```
(\LASTNODE
  [LAMBDA (TREE)                                                 ; Edited 14-Apr-93 16:29 by jds
    (LET ((COUNT (fetch (BTREENODE COUNT) of TREE))
          CHILD)
         (for ITEM# from (LLSH (IDIFFERENCE COUNT 1)
                               2)
            to 0 by -4 when (SETQ CHILD (\GETBASEPTR TREE ITEM#)) do (RETURN (COND
                                                                               ((type? BTREENODE CHILD)
                                                                                (\LASTNODE CHILD))
                                                                               (T TREE])
```

## (\MATCHPCS

```
(\MATCHPCS
  [LAMBDA (PCNODE)                                               ; Edited  5-May-93 17:57 by jds
    ;; Make sure that any pieces pointed to this node point back to this node.
    (bind PC for OFFSET from 0 to \WORDSINBTREEMAIN by 4 as I from 1 to (fetch (BTREENODE COUNT) of PCNODE)
       do (SETQ PC (\GETBASEPTR PCNODE OFFSET))
          (COND
             ((type? PIECE PC)
              (replace (PIECE PTREENODE) of PC with PCNODE))
             ((type? BTREENODE PC)
              (replace (BTREENODE UPWARD) of PC with PCNODE])
```

## (\SPLITTREE

```
(\SPLITTREE
  [LAMBDA (PCNODE)                                               ; Edited 21-Mar-95 15:26 by sybalsky:mv:envos
    ;; We're adding piece NEW in front of OLD.  OLD is represented in the B-tree node PCNODE, which is full.
```

```
;; Split PCNODE in two and propogate any changes upward.
(UNINTERRUPTABLY
    [LET ((UPWARD (fetch (BTREENODE UPWARD) of PCNODE))
          COUNT ITEM# NEW1 NEW2)
         (COND
            (UPWARD
                    ;; Easy case:  This is not the root node, so split the node and propogate up.
                    (SETQ NEW1 (create BTREENODE using PCNODE))
                    ;; Clean out upper 3 child entries, leaving only the lower 2.  Have to tell GC about actual child slots being set to NIL
                    ;; (hence \RPLPTRs):
                    (for OFST from \BTREETOPHALFOFFSET to \WORDSINBTREEMAIN by 4
                       do (\RPLPTR NEW1 OFST NIL)
                          (\PUTBASEFIXP NEW1 (IPLUS OFST 2)
                                 0))
                    (replace (BTREENODE COUNT) of NEW1 with (LRSH \BTREEMAXENTRIES 1))
                    (\TEDIT.SET-TOTLEN NEW1)
                    (\MATCHPCS NEW1)
                    ;; Now clean up the old piece, to contain only the upper 3 original children:
                    (for OFST from 0 to (SUB1 \BTREETOPHALFOFFSET) by 4 do
                                                                    ; For GC, have to tell it we've dropped pointers to first N/2 pieces
                                                                          (\RPLPTR PCNODE OFST NIL))
                    ;; Move upper N/2+1 down
                    [for OFST from 0 to \BTREETOPHALFOFFSET by 4 as UPPEROFST from \BTREETOPHALFOFFSET
                       by 4 do (\PUTBASEPTR PCNODE OFST (\GETBASEPTR PCNODE UPPEROFST))
                               (\PUTBASEFIXP PCNODE (IPLUS 2 OFST)
                                      (\GETBASEFIXP PCNODE (IPLUS 2 UPPEROFST]
                    ;; And clean out upper 2 slots, without the GC seeing it:
                    (for OFST from (IPLUS \BTREEWORDSPERENTRY \BTREETOPHALFOFFSET) to \WORDSINBTREEMAIN
                       by \BTREEWORDSPERENTRY do (\PUTBASEPTR PCNODE OFST NIL)
                                                 (\PUTBASEFIXP PCNODE (IPLUS OFST 2)
                                                        0))
                    (replace (BTREENODE COUNT) of PCNODE with (ADD1 (LRSH \BTREEMAXENTRIES 1)))
                    (\TEDIT.SET-TOTLEN PCNODE)
                    (SETQ COUNT (fetch (BTREENODE COUNT) of UPWARD))
                    (\INSERTTREE NEW1 PCNODE UPWARD NIL (fetch (BTREENODE TOTLEN) of PCNODE)))
            (T      ;; Hard case:  This is the root node.  We need to create 2 new nodes, put the split parts there, and re-use this node as the root.
                    (SETQ NEW1 (create BTREENODE using PCNODE))
                    (for OFST from \BTREETOPHALFOFFSET to \WORDSINBTREEMAIN by 4
                       do (\RPLPTR NEW1 OFST NIL)
                          (\PUTBASEFIXP NEW1 (IPLUS OFST 2)
                                 0))
                    (replace (BTREENODE UPWARD) of NEW1 with PCNODE)
                    (replace (BTREENODE COUNT) of NEW1 with (LRSH \BTREEMAXENTRIES 1))
                    (\TEDIT.SET-TOTLEN NEW1)
                    (\MATCHPCS NEW1)
                    ;; --
                    (SETQ NEW2 (create BTREENODE using PCNODE))
                    (for OFST from 0 to (SUB1 \BTREETOPHALFOFFSET) by 4 do
                                                                    ; For GC, have to tell it we've dropped pointers to first N/2 pieces
                                                                          (\RPLPTR NEW2 OFST NIL))
                    [for OFST from 0 to \BTREETOPHALFOFFSET by 4 as UPPEROFST from \BTREETOPHALFOFFSET by 4
                       do (\PUTBASEPTR NEW2 OFST (\GETBASEPTR NEW2 UPPEROFST))
                          (\PUTBASEFIXP NEW2 (IPLUS 2 OFST)
                                 (\GETBASEFIXP NEW2 (IPLUS 2 UPPEROFST]
                    (for OFST from (IPLUS \BTREEWORDSPERENTRY \BTREETOPHALFOFFSET) to \WORDSINBTREEMAIN
                       by \BTREEWORDSPERENTRY do (\PUTBASEPTR NEW2 OFST NIL)
                                                 (\PUTBASEFIXP NEW2 (IPLUS OFST 2)
                                                        0))
                    (replace (BTREENODE UPWARD) of NEW2 with PCNODE)
                    (replace (BTREENODE COUNT) of NEW2 with (ADD1 (LRSH \BTREEMAXENTRIES 1)))
                    (\TEDIT.SET-TOTLEN NEW2)
                    (\MATCHPCS NEW2)
                    ;; Now clean out the top-level node, and fill it in with its new children.
                    (for OFST from 0 to \WORDSINBTREEMAIN by \BTREEWORDSPERENTRY
                       do ;; Clean out the entries in the node, so we don't over-write them by mistake, thus losing refcount sync.
                          (\RPLPTR PCNODE OFST NIL)
                          (\PUTBASEFIXP PCNODE (IPLUS 2 OFST)
                                 0))
                    (\RPLPTR PCNODE 0 NEW1)                         ; Add first new node
                    (\PUTBASEFIXP PCNODE 2 (ffetch (BTREENODE TOTLEN) of NEW1))
                    (\RPLPTR PCNODE 4 NEW2)                         ; And the second....
                    (\PUTBASEFIXP PCNODE 6 (ffetch (BTREENODE TOTLEN) of NEW2))
                    (freplace (BTREENODE COUNT) of PCNODE with 2)
                    (freplace (BTREENODE TOTLEN) of PCNODE with (IPLUS (ffetch (BTREENODE TOTLEN) of NEW1)
                                                                       (ffetch (BTREENODE TOTLEN) of NEW2])])
```

## (\TEDIT.UPDATETREE

```
[LAMBDA (PCNODE DELTA)                                  ; Edited 21-Mar-95 14:40 by sybalsky:mv:envos
  ;; The size of the text represented by PCNODE has grown by DELTA.  Update all of PCNODE's parents to reflect the change in length.
  (LET ((UPWARD (fetch (BTREENODE UPWARD) of PCNODE)))
       (while UPWARD do ;; Keep going up in the tree til we hit the top.
                        (for old ITEM# from 0 by 4 as I from 1 to (ffetch (BTREENODE COUNT) of UPWARD)
                           when (EQ (\GETBASEPTR UPWARD ITEM#)
                                    PCNODE)
                           do (\PUTBASEFIXP UPWARD (IPLUS ITEM# 2)
                                    (IPLUS (\GETBASEFIXP UPWARD (IPLUS ITEM# 2))
                                           DELTA))
                              (add (fetch (BTREENODE TOTLEN) of UPWARD)
                                   DELTA)
                              (RETURN)
                           FINALLY (HELP "PCNODE not in upward node."))
                        (SETQ PCNODE UPWARD)
                        (SETQ UPWARD (fetch (BTREENODE UPWARD) of PCNODE])
```

## (\TEDIT.PIECE-CHNO

```
[LAMBDA (PC)
  (LET ((PCNODE (fetch (PIECE PTREENODE) of PC))
        (CHARCOUNT 0))
       (while PCNODE do [add CHARCOUNT (for OFST from 0 by 4 while (NEQ PC (\GETBASEPTR PCNODE OFST))
                                          sum (\GETBASEFIXP PCNODE (IPLUS OFST 2]
                        (SETQ PC PCNODE)
                        (SETQ PCNODE (fetch (BTREENODE UPWARD) of PCNODE)))
       (ADD1 CHARCOUNT])
```

## (\TEDIT.SET-TOTLEN

```
[LAMBDA (PCNODE)                                        ; Edited  9-May-93 15:40 by jds
  ;; Fix the TOTLEN field of a node to match the sum of its childrens' lengths
  (replace (BTREENODE TOTLEN) of PCNODE with (for I from 1 to (fetch (BTREENODE COUNT) of PCNODE) as ITEM#
                                                from 2 by 4 sum (\GETBASEFIXP PCNODE ITEM#])

)

(DEFINEQ
```

## (DISPTREE

```
[LAMBDA (TREE DEPTH)                                    ; Edited 13-Apr-90 15:00 by ON
  (LET [(G (TREEGRAPHNODE TREE NIL (OR (NUMBERP DEPTH)
                                       T]
       (SHOWGRAPH (LAYOUTGRAPH (CADR G)
                               (LIST (CAR G))
                               '(VERTICAL))
             NIL
             #'(LAMBDA (X)
                 (INSPECT (fetch NODEID of X])
```

## (TREEGRAPHNODE

```
[LAMBDA (TREE PARENT DEPTH)                             ; Edited 12-Jun-90 10:33 by mitani
  (LET (THISNODE NEWDEPTH NODEID LONODES HINODES BFNODE BFNODEID RANKNODE RANKNODEID)
       (COND
          ((ATOM TREE)
           (LIST [fetch NODEID of (SETQ THISNODE (NODECREATE (CONS)
                                                             TREE NIL NIL (LIST PARENT]
                 (LIST THISNODE)))
          ((OR (EQ DEPTH T)
               (AND (NUMBERP DEPTH)
                    (>= DEPTH 0)))
           (SETQ NEWDEPTH (COND
                             ((NUMBERP DEPTH)
                              (SUB1 DEPTH))
                             (T DEPTH)))
           (SETQ NODEID (fetch (PCTNODE PCE) of TREE))
           (SETQ LONODES (TREEGRAPHNODE (fetch (PCTNODE LO) of TREE)
                                        NODEID NEWDEPTH))
           (SETQ HINODES (TREEGRAPHNODE (fetch (PCTNODE HI) of TREE)
                                        NODEID NEWDEPTH))
           (SETQ BFNODE (NODECREATE (SETQ BFNODEID (CONS))
                                    (fetch (PCTNODE BF) of TREE)
                                    NIL NIL (LIST NODEID)))
           (SETQ RANKNODE (NODECREATE (SETQ RANKNODEID (CONS))
                                      (fetch (PCTNODE RANK) of TREE)
                                      NIL NIL (LIST NODEID)))
           [SETQ THISNODE (NODECREATE NODEID (fetch (PCTNODE CHNUM) of TREE)
                                      NIL
                                      (LIST (CAR LONODES)
                                            BFNODEID RANKNODEID (CAR HINODES))
                                      (AND PARENT (LIST PARENT]
           (LIST (fetch NODEID of THISNODE)
```

```
                    (APPEND (LIST THISNODE BFNODE RANKNODE)
                            (CADR LONODES)
                            (CADR HINODES])

)

(DECLARE%: EVAL@COMPILE

(DATATYPE BTREENODE (;; An order-4 BTREE node for representing the piece table for TEdit.
                     DOWN1
                     (DLEN1 FIXP)
                     DOWN2
                     (DLEN2 FIXP)
                     DOWN3
                     (DLEN3 FIXP)
                     DOWN4
                     (DLEN4 FIXP)
                     DOWN5
                     (DLEN5 FIXP)
                     DOWN6
                     (DLEN6 FIXP)
                     DOWN7
                     (DLEN7 FIXP)
                     DOWN8
                     (DLEN8 FIXP)
                     SPARE5                                  ; Used only to hold the extra piece when we're overflowing
                     (SPARELEN FIXP)                         ; So the code is easy and fast.
                     (COUNT BITS 4)                          ; # of children of this node
                     (UPWARD XPOINTER)                       ; Parent of this node, if any.
                     (TOTLEN FIXP)                           ; Total length of this tree and subtrees

                     ))
)

(/DECLAREDATATYPE 'BTREENODE
       '(POINTER FIXP POINTER FIXP POINTER FIXP POINTER FIXP POINTER FIXP POINTER FIXP POINTER FIXP POINTER FIXP
                 POINTER FIXP (BITS 4)
                 XPOINTER FIXP)
       ;; ---field descriptor list elided by lister---
       '40)

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

(ADDTOVAR NLAMA )

(ADDTOVAR NLAML )

(ADDTOVAR LAMA )
)

(PUTPROPS PCTREE COPYRIGHT ("Venue & Xerox Corporation" 1990 1991 1993 1994 1995))
```

## FUNCTION INDEX



## CONSTANT INDEX



## RECORD INDEX