```
(RPAQQ READNUMBERCOMS
       [(FNS \NUMBERPAD.READER.CLOSEFN \READNUMBER.FLASHAREA RNUMBER NUMBERPAD.READ NUMBERPAD.READER.HANDLE.CHAR
             NUMBERPAD.READER.DECODE CREATE.NUMBERPAD.READER BREAK.MSG.INTO.LINES REGIONONSCREEN
             DISPLAY/NUMBER/READER/TOTAL NUMBER.READER.HANDLER NUMBERPAD.HELDFN \READNUMBER.OUTLINEREGION)
        (UGLYVARS RNUMBER.CLEAR.BITMAP RNUMBER.ABORT.BITMAP)
        (COMS (* stuff to dummy up a definition of TEDIT.GETSYNTAX if it isn't defined.)
              (INITVARS (TEDIT.READTABLE T))
              (P (MOVD? 'GETSYNTAX 'TEDIT.GETSYNTAX])

(DEFINEQ

(\NUMBERPAD.READER.CLOSEFN
  [LAMBDA (WINDOW)
    (WINDOWPROP WINDOW 'FINISHEDFLG 'ABORT])


(\READNUMBER.FLASHAREA
  [LAMBDA (LFT BTM WDTH HGHT WIN)                                    (* rrb "28-JUN-82 19:17")
                                                                    (* flashes a region of a window.)
    (BITBLT NIL NIL NIL WIN LFT BTM WDTH HGHT 'TEXTURE 'INVERT BLACKSHADE)
    (DISMISS 60)
    (BITBLT NIL NIL NIL WIN LFT BTM WDTH HGHT 'TEXTURE 'INVERT BLACKSHADE])


(RNUMBER
  [LAMBDA (MSG POSITION MSGFONT DIGITFONT INCLUDEABORTFLG FLOATINGPTFLG POSITIVEONLYFLG ACCEPTTYPEINFLG)
                                                                    (* rrb "14-May-86 19:02")
                                                                    (* creates a numberpad window menu and lets the user enter a
    number.)                                                        (* it is substantially more efficient to save the
                                                                    NUMBERPAD/READER and call NUMBERPAD.READ directly.)
    (NUMBERPAD.READ (CREATE.NUMBERPAD.READER MSG POSITION MSGFONT DIGITFONT INCLUDEABORTFLG FLOATINGPTFLG
                        POSITIVEONLYFLG)
          ACCEPTTYPEINFLG])


(NUMBERPAD.READ
  [LAMBDA (NUMBERPAD/READER ACCEPTTYPEINFLG)                        ; Edited 24-May-93 23:40 by sybalsky:mv:envos

    ;; allows the user to enter a number with the numberpad.  if ACCEPTTYPEINFLG is T, it also allows the user to type in numbers as well.

    ;; position the number pad near the current reading location.

    (WINDOWPROP NUMBERPAD/READER 'TOTAL 0)                          ; start a mouse process in case this one is it.
    (SPAWN.MOUSE)                                                   ; make sure the finished flag is initialized.
    (WINDOWPROP NUMBERPAD/READER 'FINISHEDFLG NIL)                  ; fix it so that closing the window does an abort.
    (WINDOWADDPROP NUMBERPAD/READER 'CLOSEFN (FUNCTION \NUMBERPAD.READER.CLOSEFN))
    (RESETLST
        [COND
            (ACCEPTTYPEINFLG (RESETSAVE (TTYDISPLAYSTREAM NUMBERPAD/READER))
                   (RESETSAVE (TTY.PROCESS (THIS.PROCESS)))
                   (RESETSAVE (CONTROL T))
                   (RESETSAVE (ECHOMODE NIL]
        (RESETSAVE (OPENW NUMBERPAD/READER)
               (LIST 'CLOSEW NUMBERPAD/READER))
        (DISPLAY/NUMBER/READER/TOTAL NUMBERPAD/READER)             ; wait for the menu handler to set that it is finished.
        [bind FINISHVAL until (SETQ FINISHVAL (WINDOWPROP NUMBERPAD/READER 'FINISHEDFLG NIL))
           do                                                      ; keep bringing the numberpad to the top.
              (TOTOPW NUMBERPAD/READER)
              (COND
                 ((AND ACCEPTTYPEINFLG (HASTTYWINDOWP)
                       (READP T))                                  ; this process has the tty and the user typed a character
                     (NUMBERPAD.READER.HANDLE.CHAR (NUMBERPAD.READER.DECODE (READC T))
                            NUMBERPAD/READER)))
              (DISMISS 100)
           finally                                                ; remove the closefn so that it doesn't get run on the way out.
              (WINDOWDELPROP NUMBERPAD/READER 'CLOSEFN (FUNCTION \NUMBERPAD.READER.CLOSEFN))
              (RETURN (COND
                          ((EQ FINISHVAL 'ABORT)

                    ;; means the numberpad reader was closed.  If the number pad includes the ABORT command, do what it would
                    ;; do, otherwise the program is not expecting NIL so cause an error.

                           (COND
                              ([MEMBER '% (fetch (MENU ITEMS) of (CAR (WINDOWPROP NUMBERPAD/READER 'MENU]
```

```
                                                                    ; no ABORT command
                                     (ERROR!))
                                    (T NIL)))
                         (T (WINDOWPROP NUMBERPAD/READER 'TOTAL])])
```

(**NUMBERPAD.READER.HANDLE.CHAR**
```
  [LAMBDA (DIGIT WIN)                                               ; Edited 19-Jan-89 17:50 by gadener
                                                                    (* handles a key stroke or menu digit selection for a number pad
     reader.)
     (PROG (TOTAL POWER)
           (SETQ TOTAL (WINDOWPROP WIN 'TOTAL))
           [WINDOWPROP WIN 'TOTAL
                  (SELECTQ DIGIT
                      (bs (COND
                             [(SETQ POWER (WINDOWPROP WIN 'DECIMALPOWER))
                                                                    (* have read decimal pt -
                                                                    much harder)
                               (COND
                                  ((EQ POWER 1)                     (* backspace over the decimal point.)
                                   (WINDOWPROP WIN 'DECIMALPOWER NIL)
                                   (FIX TOTAL))
                                  (T (WINDOWPROP WIN 'DECIMALPOWER (SETQ POWER (SUB1 POWER)))
                                                                    (* dirty but effective.)
                                       (PROG ((TOTSTR (MKSTRING TOTAL)))
                                                                    (* SUBSTRING will be NIL if the total has a trailing zero.)
                                              (RETURN (MKATOM (OR (SUBSTRING TOTSTR 1 (PLUS (STRPOS "." TOTSTR)
                                                                                       (SUB1 POWER)))
                                                                 TOTSTR]
                              (T                                    (* no decimal point)
                                 (IQUOTIENT TOTAL 10))))
                      (- (MINUS TOTAL))
                      (%                                            (* empty key)
                          TOTAL)
                      (%.                                           (* decimal point)
                          (COND
                             ((WINDOWPROP WIN 'DECIMALPOWER)        (* already has a decimal pt, don't do anything)
                              (RETURN))
                             (T (WINDOWPROP WIN 'DECIMALPOWER 1)
                                (FLOAT TOTAL))))
                      (ok (WINDOWPROP WIN 'FINISHEDFLG T)
                          (RETURN))
                      (COND
                         ((OR (EQ DIGIT RNUMBER.ABORT.BITMAP)
                              (EQ DIGIT 'abt))                      (* abort key)
                          (WINDOWPROP WIN 'TOTAL NIL)
                          (WINDOWPROP WIN 'FINISHEDFLG T)
                          (RETURN))
                         ((OR (EQ DIGIT RNUMBER.CLEAR.BITMAP)
                              (EQ DIGIT 'clr))                      (* clear key)
                          (WINDOWPROP WIN 'DECIMALPOWER NIL)
                          0)
                         ((EQ (WINDOWPROP WIN 'MAXDIGITS)
                              (NCHARS (ABS TOTAL)))                 (* don't take any more.)
                          (\READNUMBER.FLASHAREA 0 0 1000 1000 WIN)
                          TOTAL)
                         [(NUMBERP DIGIT)
                          (COND
                             [(SETQ POWER (WINDOWPROP WIN 'DECIMALPOWER))
                                                                    (* have read decimal pt)
                              (WINDOWPROP WIN 'DECIMALPOWER (ADD1 POWER))
                              (COND
                                 [(GEQ TOTAL 0)
                                  (PLUS TOTAL (FQUOTIENT DIGIT (EXPT 10 POWER]
                                 (T (DIFFERENCE TOTAL (FQUOTIENT DIGIT (EXPT 10 POWER]
                             ((GEQ TOTAL 0)
                              (PLUS (TIMES TOTAL 10)
                                    DIGIT))
                             (T (DIFFERENCE (TIMES TOTAL 10)
                                    DIGIT]
                         (T                                        (* uninteresting key struck, ignore it)
                            (RETURN]
           (DISPLAY/NUMBER/READER/TOTAL WIN])
```

(**NUMBERPAD.READER.DECODE**
```
  [LAMBDA (CHAR)                                                   (* rrb "14-May-86 18:49")
                                                                   (* decodes a keystroke into the corresponding number pad
                                                                   reader menu item.)

     (OR (NUMBERP CHAR)
         (SELECTQ (TEDIT.GETSYNTAX CHAR TEDIT.READTABLE)
             (CHARDELETE 'bs)
             ((WORDDELETE LINEDELETE)
                  'clr)
             (DELETE 'abt)
             (COND
                ((OR (EQ CHAR '%
```

```
)
                            (EQ CHAR '%
))
                  'ok)
                (T CHAR])
```

(**CREATE.NUMBERPAD.READER**
```
  [LAMBDA (MSG WPOSITION MSGFONT DIGITFONT INCLUDEABORTFLG FLOATINGPTFLG POSITIVEONLYFLG)
```
                                                          (* rrb "14-May-86 18:57")

            (* creates a window menu that displays the digits in a numberpad and lets the user enter a number.
            It also includes a backspace and a enter)

```
     (RESETFORM (RADIX 10)
          (PROG ((NUMBER/READER/MAXDIGITS (COND
                                            (FLOATINGPTFLG 8)
                                            (T 6)))
                 WIN READERWIDTH PADLEFT TOTALREGION (DIGITFONT (OR DIGITFONT (FONTCREATE BOLDFONT)))
                 (MSGFONT (OR MSGFONT (FONTCREATE DEFAULTFONT)))
                 NUMBERPAD TOTALWIDTH FONTHEIGHT MSGLINES)
               [SETQ NUMBERPAD (create MENU
                                       ITEMS _
                                       [CONS (COND
                                               [POSITIVEONLYFLG (COND
                                                                  ((AND INCLUDEABORTFLG FLOATINGPTFLG)
                                                          (* no other place to put the backspace)
                                                                   'bs)
                                                                  (T '% ]
                                               (T '-))
                                             (APPEND [COND
                                                       [INCLUDEABORTFLG
                                                        (COND
                                                          ((AND (IGREATERP (SETQ FONTHEIGHT
                                                                            (FONTPROP MSGFONT 'HEIGHT))
                                                                   10)
                                                             (ILESSP FONTHEIGHT 20))
                                                          (* only use the bitmap for fonts near 10 or 12.0)
                                                           (LIST RNUMBER.ABORT.BITMAP RNUMBER.CLEAR.BITMAP))
                                                          (T '(abt clr]
                                                       (FLOATINGPTFLG
                                                          (* put backspace in top row because decimal pt appears in the
                                                          bottom.)
                                                          '(bs clr))
                                                       (T '(%  clr]
                                                     '(1 2 3 4 5 6 7 8 9)
                                                     [COND
                                                       (FLOATINGPTFLG
                                                          (* if floating point numbers are ok, include decimal point.)
                                                          '(%.))
                                                       (T '(bs]
                                                     '(0 ok]
                                       MENUCOLUMNS _ 3
                                       CENTERFLG _ T
                                       MENUFONT _ DIGITFONT
                                       WHENHELDFN _ (FUNCTION NUMBERPAD.HELDFN)
                                       WHENSELECTEDFN _ (FUNCTION NUMBER.READER.HANDLER)
                                       MENUOUTLINESIZE _ 2
                                       ITEMHEIGHT _ (IPLUS 2 (FONTPROP DIGITFONT 'HEIGHT]
                                                          (* leave room for three lines at the top and the number at the
                                                          left.)
               (SETQ WIN (CREATEW [REGIONONSCREEN WPOSITION
                                     [WIDTHIFWINDOW (SETQ READERWIDTH
                                                      (IPLUS [SETQ PADLEFT
                                                               (IPLUS 12 (SETQ TOTALWIDTH
                                                                           (ITIMES (ADD1
                                                                                     NUMBER/READER/MAXDIGITS
                                                                                     )
                                                                             (CHARWIDTH (CHARCODE 0)
                                                                                     DIGITFONT]
                                                        (fetch (MENU IMAGEWIDTH) of NUMBERPAD]
                                     (HEIGHTIFWINDOW (IPLUS (COND
                                                              [MSG
                                                          (* if there is a msg, leave room for it at the top.)
                                                               (ITIMES (LENGTH (SETQ MSGLINES
                                                                                (BREAK.MSG.INTO.LINES
                                                                                 MSG MSGFONT
                                                                                 READERWIDTH)))
                                                                 (FONTPROP MSGFONT 'HEIGHT]
                                                              (T 0))
                                                       (fetch (MENU IMAGEHEIGHT) of NUMBERPAD]
                                     NIL NIL T))
               [COND
                 (MSG                                                (* if there is a msg, print it at the top.)
                      (DSPFONT MSGFONT WIN)
                      (MOVETOUPPERLEFT WIN)
                      (for LINE in MSGLINES do (PRIN3 LINE WIN)
```

```
                                                                    (TERPRI WIN]
                    (OPENW WIN)

        (* window is opened because of bug in ADDMENU that it doesn't work unless window is open.)

                    (ADDMENU NUMBERPAD WIN (create POSITION
                                                    XCOORD _ PADLEFT
                                                    YCOORD _ 0))
                    [WINDOWPROP WIN 'TOTALREG (SETQ TOTALREGION (create REGION
                                                                    LEFT _ 6
                                                                    BOTTOM _ (IQUOTIENT (fetch (MENU IMAGEHEIGHT)
                                                                                            of NUMBERPAD)
                                                                                2)
                                                                    WIDTH _ TOTALWIDTH
                                                                    HEIGHT _ (FONTPROP DIGITFONT 'HEIGHT]
                    (\READNUMBER.OUTLINEREGION TOTALREGION WIN 2)
                    (DSPFONT DIGITFONT WIN)
                    (WINDOWPROP WIN 'TOTAL 0)
                    (WINDOWPROP WIN 'MAXDIGITS NUMBER/READER/MAXDIGITS)
                    (DISPLAY/NUMBER/READER/TOTAL WIN)
                    (CLOSEW WIN)
                    (RETURN WIN])
```

(BREAK.MSG.INTO.LINES
```
  [LAMBDA (MSG FONT WIDTH)                                             (* rrb "27-Aug-85 14:35")
                                                                      (* returns a list of string that will fit in WIDTH if printed in FONT)

    (PROG ([MSGSTR (COND
                        ((LISTP MSG)                                   (* remove the outer parens)
                         (SUBSTRING (MKSTRING MSG)
                                    2 -2))
                        (T (MKSTRING MSG]
            (THISLINE 0)
            (BEGIN 1)
            LASTSPACE STRLST CHARWIDTH CHARCODE)
        [for I from 1 to (NCHARS MSGSTR) do (SETQ CHARWIDTH (CHARWIDTH (SETQ CHARCODE (NTHCHARCODE MSGSTR I))
                                                                        FONT))
                                            (COND
                                                [(GREATERP (SETQ THISLINE (IPLUS THISLINE CHARWIDTH))
                                                        WIDTH)      (* this character would go past)
                                                (COND
                                                    ((EQ CHARCODE (CHARCODE SPACE))
                                                    [SETQ STRLST (NCONC1 STRLST (SUBSTRING MSGSTR BEGIN
                                                                                    (SUB1 I]
                                                    (SETQ THISLINE 0)
                                                    (SETQ BEGIN (ADD1 I))
                                                    (SETQ LASTSPACE))
                                                    (LASTSPACE     (* this line has a space in it.)
                                                            [SETQ STRLST (NCONC1 STRLST (SUBSTRING MSGSTR BEGIN
                                                                                            (SUB1 LASTSPACE]
                                                            (SETQ BEGIN (ADD1 LASTSPACE))
                                                            (SETQ THISLINE 0)
                                                            (for NL from (ADD1 LASTSPACE) to I
                                                                do (SETQ THISLINE (IPLUS (CHARWIDTH (NTHCHARCODE
                                                                                                        MSGSTR NL)
                                                                                            FONT)
                                                                                    THISLINE)))
                                                            (SETQ LASTSPACE))
                                                    (T              (* this line doesn't have a space)
                                                        [SETQ STRLST (NCONC1 STRLST (SUBSTRING MSGSTR BEGIN
                                                                                        (SUB1 I]
                                                        (SETQ BEGIN I)
                                                        (SETQ THISLINE CHARWIDTH]
                                                ((EQ CHARCODE (CHARCODE SPACE))
                                                                (* note the position of the space)
                                                (SETQ LASTSPACE I)))
                finally (COND
                            ((GREATERP (SUB1 I)
                                    BEGIN)
                            (SETQ STRLST (NCONC1 STRLST (SUBSTRING MSGSTR BEGIN -1]
            (RETURN STRLST])
```

(REGIONONSCREEN
```
  [LAMBDA (POS WIDTH HEIGHT)                                           (* rrb "20-May-86 11:42")

        (* returns the region WIDTH by HEIGHT that is nearest to POS or the cursor position while still being on the screen.)

    (PROG (LEFT BOTTOM)
        (COND
            ((POSITIONP POS)
            (SETQ LEFT (IMAX (fetch (POSITION XCOORD) of POS)
                            0))
            (SETQ BOTTOM (IMAX (fetch (POSITION YCOORD) of POS)
                            0)))
            (T (GETMOUSESTATE)
                (SETQ LEFT LASTMOUSEX)
```

```
                (SETQ BOTTOM LASTMOUSEY)))
          [COND
             ((IGREATERP (IPLUS LEFT WIDTH)
                    SCREENWIDTH)
              (SETQ LEFT (IMAX 0 (IDIFFERENCE SCREENWIDTH WIDTH]
          [COND
             ((IGREATERP (IPLUS BOTTOM HEIGHT)
                    SCREENHEIGHT)
              (SETQ BOTTOM (IMAX 0 (IDIFFERENCE SCREENHEIGHT HEIGHT]
          (RETURN (create REGION
                      LEFT _ LEFT
                      BOTTOM _ BOTTOM
                      WIDTH _ WIDTH
                      HEIGHT _ HEIGHT])
```

(**DISPLAY/NUMBER/READER/TOTAL**
```
  [LAMBDA (WIN)                                        (* rrb "14-May-86 18:36")
                                                       (* displays the number total in the box in the window.)
     (PROG [(TOTALREG (WINDOWPROP WIN 'TOTALREG))
            (DECIMALPLACES (WINDOWPROP WIN 'DECIMALPOWER]
        (DSPFILL TOTALREG WHITESHADE 'REPLACE WIN)
        (RESETFORM (RADIX 10)
               (CENTERPRINTINREGION
                 [COND
                    [DECIMALPLACES
```

(* printing a decimal number must check to make sure the correct number of decimal places print.)

```
                              (PROG ([TOTSTR (MKSTRING (WINDOWPROP WIN 'TOTAL]
                                      DECPOS NAFTERDEC NCHARS)
                                 (SETQ NCHARS (NCHARS TOTSTR))
                                 (SETQ DECPOS (STRPOS "." TOTSTR))
                                 (RETURN (COND
                                            ((EQ (SUB1 DECIMALPLACES)
                                                 (SETQ NAFTERDEC (DIFFERENCE NCHARS DECPOS)))
                                                                 (* right number of places)
                                              TOTSTR)
                                            [(GEQ NAFTERDEC DECIMALPLACES)
                                                                 (* strip off the unwanted ones.)
                                              (SUBSTRING TOTSTR 1 (PLUS DECPOS (SUB1 DECIMALPLACES]
                                            (T                   (* not enough zeros on the end)
                                               (CONCAT TOTSTR (bind STR for I from 1
                                                                to (DIFFERENCE (SUB1 DECIMALPLACES)
                                                                           NAFTERDEC)
                                                              do (COND
                                                                     (STR (SETQ STR (CONCAT STR "0")))
                                                                     (T (SETQ STR "0")))
                                                              finally (RETURN STR]
                    (T (WINDOWPROP WIN 'TOTAL]
                 TOTALREG WIN])
```

(**NUMBER.READER.HANDLER**
```
  [LAMBDA (DIGIT MENU BUTTON)                           (* rrb "14-May-86 15:37")
```

(* selected fn for a numberpad reader. adds the digit to the current total and updates the display.)

```
     (NUMBERPAD.READER.HANDLE.CHAR DIGIT (WFROMMENU MENU])
```

(**NUMBERPAD.HELDFN**
```
  [LAMBDA (ITEM MENU BUTTON)                            (* rrb "26-Aug-85 13:50")
                                                        (* prints the help information for a numberpad.)
     (PROMPTPRINT (SELECTQ ITEM
                     (bs "Will erase the last digit entered.")
                     (ok "Indicates that you are through entering the number.")
                     (clr "Will reset the total to 0")
                     (abt "will abort this question.")
                     (- " will change the sign of the total")
                     (%. "will enter a decimal point.")
                     (% "doesn't do anything.")
                     (COND
                        ((EQ ITEM RNUMBER.ABORT.BITMAP)        (* abort bitmap)
                         "will abort this question.")
                        ((EQ ITEM RNUMBER.CLEAR.BITMAP)        (* abort bitmap)
                         "Will reset the total to 0")
                        ((NLISTP ITEM)
                         "Will put this digit on the right of the total."])
```

(**READNUMBER.OUTLINEREGION**
```
  [LAMBDA (REG WIN OUTLINESIZE)                         ; Edited 12-Jun-90 10:55 by mitani
                                                        (* puts a black outline around REG.)
     (PROG ((N (OR (FIXP OUTLINESIZE)
                2)))
           (BITBLT NIL NIL NIL WIN (IDIFFERENCE (fetch (REGION LEFT) of REG)
```

```
                                          N)
                  (IDIFFERENCE (fetch (REGION BOTTOM) of REG)
                          N)
                  (IPLUS (fetch (REGION WIDTH) of REG)
                         (ITIMES N 2))
                  (IPLUS (fetch (REGION HEIGHT) of REG)
                         (ITIMES N 2))
                  'TEXTURE
                  'REPLACE BLACKSHADE)
           (BITBLT NIL NIL NIL WIN (fetch (REGION LEFT) of REG)
                  (fetch (REGION BOTTOM) of REG)
                  (fetch (REGION WIDTH) of REG)
                  (fetch (REGION HEIGHT) of REG)
                  'TEXTURE
                  'REPLACE
                  (DSPTEXTURE NIL WIN])
)

(READVARS-FROM-STRINGS '(RNUMBER.CLEAR.BITMAP RNUMBER.ABORT.BITMAP)
       "({(READBITMAP)(14 14
       %"GJ@@%"
       %"DJCL%"
       %"DBBD%"
       %"DBCL%"
       %"DJB@%"
       %"GKKL%"
       %"@@@@%"
       %"@@@@%"
       %"@GKL%"
       %"@DJD%"
       %"@GKL%"
       %"@DJH%"
       %"@DJD%"
       %"@@@@%")}  {(READBITMAP)(14 14
       %"GKL@%"
       %"DJD@%"
       %"DKL@%"
       %"GJD@%"
       %"DJD@%"
       %"DKL@%"
       %"@@@@%"
       %"@@CH%"
       %"GKM@%"
       %"DJE@%"
       %"DKM@%"
       %"DJI@%"
       %"GJM@%"
       %"@@@@%")})
       ")
```

          (* * stuff to dummy up a definition of TEDIT.GETSYNTAX if it isn't defined.)

(RPAQ? **TEDIT.READTABLE** T)

(MOVD? 'GETSYNTAX 'TEDIT.GETSYNTAX)

(PUTPROPS **READNUMBER COPYRIGHT** ("Venue & Xerox Corporation" 1982 1983 1984 1985 1986 1989 1990 1993))

FUNCTION INDEX

VARIABLE INDEX