```
; ============================================================
;  geoGopherTxt: view text file or info items in a scroll pane
; ============================================================
.if             Pass1
                .noeqin
                .include    geoGopherSym
                .include    geoGopherMac
                .include    geoGopher.inc
                .include    ultimate.inc
                .eqin
.endif
; ============================================================
;  Show text or info items in a dialog box. Called from dialog
;  (vwTextDB) via DB_USR_ROUT.
;               pass:       a0, address of info item(s) or loaded text
; ============================================================
showText:       LoadB       r3L,7
                LoadB       r3H,194
                LoadW       r4,308
                lda         #$ff            ;solid line
                jsr         VerticalLine    ;scroll bar boundary
                LoadW       r0,txtTpArw
                LoadB       r1L,38          ;fake the arrow icons
                LoadB       r1H,6           ;so we don't have to wait
                LoadB       r2L,2           ;to see them
                LoadB       r2H,8
                jsr         BitmapUp
                LoadW       r0,txtBtArw
                LoadB       r1L,38          ;X pos. (cards)
                LoadB       r1H,188         ;Y pos. (pixels)
                LoadB       r2L,2           ;width (cards)
                LoadB       r2H,8           ;height (pixels)
                jsr         BitmapUp
                MoveW       fontLoad,r0
                jsr         LoadCharSet
                LoadW       r0,fmtMsg       ;"formatting text..."
                LoadW       r11,TEXT_L
                LoadB       r1H,190
                jsr         PutString
                jsr         initText
                lda         #0
                sta         pageNdx
                sta         topItem         ;for shared code
                jsr         viewText
                lda         #0              ;clear
                jsr         SetPattern
                LoadB       r2L,185
                LoadB       r2H,193
                LoadW       r3,TEXT_L
                LoadW       r4,TEXT_L+80
                jsr         Rectangle       ;clear message
                jsr         UseSystemFont
                rts                         ;now icons will draw for real
```

```
; ============================================================
;  Initialize line and page pointers for viewing text.
;                   pass:       textType, TXT_INFO or TXT_FILE
;                               a0, start of text
; ============================================================
initText:       ldy         #0
                sty         pageNdx
                tya
10$             sta         pagePtrs,y
                iny
                cpy         #MAX_PAGE*2
                bne         10$
                ldy         pageNdx
20$             tya
                asl         a
                tay
                lda         a0L
                sta         pagePtrs,y
                lda         a0H
                sta         pagePtrs+1,y
                jsr         getLines        ;sets a0 to next page
                bcs         30$             ;end of text?
                inc         pageNdx
                ldy         pageNdx
                cpy         #MAX_PAGE
                bcc         20$
                dec         pageNdx
30$             ldy         pageNdx
                iny
                sty         numItems
                jsr         setText
                rts
```

```
; ============================================================
;  Get line pointers for a scroll pane of text.
;               pass:       a0, address of page
;                           pageNdx, current page number
;               return:     a0, address of next page
;                           carry set if end of text reached, clear otherwise
;               destroyed:  a1
; ============================================================
getLines:       lda         #MAX_LINE       ;clear line table
                asl         a
                tax
                lda         #0
10$             sta         linePtrs-1,x
                dex
                bpl         10$
                MoveW       a0,linePtrs     ;set first line index
                ldy         #0
                sty         lineNdx
reLine:         lda         (a0),y
                pha
                jsr         ckEol
                pla                         ;won't touch carry flag
                bcc         20$
                ldx         textType
                cpx         #TXT_INFO
                bne         10$             ;end of line
                cmp         #9              ;tab: end of chained info items
                bne         10$
                tya
                clc
                adc         a0L
                sta         textEnd
                lda         a0H
                adc         #0
                sta         textEnd+1
                sec
                rts
10$             iny
                jmp         nextLine
20$             iny                         ;not an EOL character
                cpy         #TEXT_WD        ;reached max. chars. across?
                bcc         reLine          ;if so, fall through
30$             tya                         ;.Y points to char that didn't fit
                pha
40$             dey
                beq         50$             ;start of line, force break
                lda         (a0),y
                cmp         #' '            ;try to break line
                bne         40$
                pla                         ;discard saved .Y
                iny                         ;point to char after blank
                bne         nextLine
50$             pla
                tay                         ;restore index to char that didn't fit
```

```
nextLine:   tya                 ;.Y points to next line
            clc
            adc     a0L
            sta     a0L
            lda     a0H
            adc     #0
            sta     a0H
            ldx     textType
            cpx     #TXT_FILE
            bne     10$
            CmpW    a0,textEnd
            bcc     10$
            rts
10$         inc     lineNdx
            lda     lineNdx
            cmp     #MAX_LINE
            bcc     20$
            clc
            rts
20$         asl     a
            tax
            lda     a0L
            sta     linePtrs,x
            lda     a0H
            sta     linePtrs+1,x
            ldy     #0
            jmp     reLine
```

```
; ============================================================
;  Check for end-of-line character.
;               pass:   character in .A
;                       .Y, pointer to character
;               return: carry set if end of line, clear otherwise
;                       .A, end-of-line character
;                       .Y, pointer to end-of-line character
;  Note that .Y may change (e.g. $0d, $0a).
;  a0 may also change if many info items are chained together.
; ============================================================
ckEol:          beq     70$
                ldx     textType
                cpx     #TXT_INFO
                bne     10$
                cmp     #9              ;tab (end of chained info items)
                beq     70$
10$             cmp     #$0d            ;DOS/MAC
                bne     30$
                iny
20$             lda     (a0),y
                cmp     #$0a            ;DOS
                beq     70$
                dey
                bne     70$
30$             cmp     #$0a            ;UNIX
                beq     70$
                cmp     #$a0            ;padding
                bne     60$
40$             iny
                bne     50$
                inc     a0H
50$             lda     (a0),y
                cmp     #$a0
                beq     40$
                dey
                cpy     #$ff
                bne     70$
                dec     a0L
                bne     70$
60$             clc                     ;not end of line
                rts
70$             sec                     ;end of line
                rts
```

```
; ============================================================
;  View a page of text (from file or gopher items) in a scroll pane.
;  Must initialize pagePtrs with initText before calling.
;                 pass:       pageNdx, page number
;                 destroyed:  a1, a2
; ============================================================
viewText:       jsr     setVwScr        ;set up scroll bar for text
                ldy     pageNdx
                tya
                asl     a
                tay
                lda     pagePtrs,y
                sta     a0L
                lda     pagePtrs+1,y
                sta     a0H             ;a0 now holds page address
                PushW   a0L
                jsr     getLines
                PopW    a0L
                LoadB   r1H,TEXT_TOP
                LoadW   r11,TEXT_L
                ldx     #0
                stx     lineNdx
viewLine:       txa
                asl     a
                tax
                lda     linePtrs,x
                sta     a0L
                lda     linePtrs+1,x
                sta     a0H
                ora     a0L             ;short page (not full)?
                bne     5$
                rts                     ;yes, must be at end
5$              lda     linePtrs+2,x    ;to check for end of line
                sta     a2L
                lda     linePtrs+3,x
                sta     a2H
                lda     a2L
                ora     a2H             ;past end of page?
                bne     15$
                ldx     pageNdx         ;yes, use start of next page
                inx
                cpx     numItems        ;number of pages
                bcs     10$             ;past last page?
                txa                     ;no, use start of next page
                asl     a
                tax
                lda     pagePtrs,x
                sta     a2L
                lda     pagePtrs+1,x
                sta     a2H
                bne     15$
10$             MoveW   textEnd,a2      ;past last page, use end marker
```

```
15$             ldy     #0
20$             lda     (a0),y
                beq     40$             ;end of text file marker
                cmp     #$0d
                beq     40$
                cmp     #$0a
                beq     40$
                cmp     #9              ;tab
                beq     40$
                cmp     #$a0            ;info padding
                beq     40$
                tax
                tya
                pha
                txa
                jsr     SmallPutChar
                pla                     ;check for end of line
                pha
                tay
                iny                     ;next char.
                tya
                clc
                adc     a0L
                sta     a1L
                lda     a0H
                adc     #0
                sta     a1H             ;pointer to next char.
                CmpW    a1,a2           ;next char. on new line?
                bcc     30$             ;nope
                pla                     ;discard
                bcs     40$
30$             pla
                tay
                iny
                bne     20$
                inc     a0H
                bne     20$
;               ================================================
40$             inc     lineNdx         ;end of line
                ldx     lineNdx
                cpx     #MAX_LINE
                bne     50$
                rts
50$             lda     r1H
                clc
                adc     #8              ;font height
                sta     r1H
                LoadW   r11,TEXT_L
                jmp     viewLine
```

```
; ============================================================
;  Set up scroll bar for text display.
; ============================================================
setVwScr:       jsr     doThumb
                lda     numItems        ;pages
                cmp     #2
                bcc     20$
                LoadW   pageDown,txtPgDn
                LoadW   pageUp,txtPgUp
                php
                sei
                LoadW   otherPressVector,chkMouse
                LoadW   topDspch,txtTop
                LoadW   botDspch,txtBot
                plp
                bra     30$
20$             php
                sei
                LoadW   otherPressVector,0
                plp
30$             rts
; ============================================================
;  custom icon handlers for view/save dialog
; ============================================================
doView:         lda     #VIEW           ;custom icon ID
                sta     sysDBData
                jmp     RstrFrmDlg
; ============================================================
doSave:         lda     #SAVE           ;custom icon ID
                sta     sysDBData
                jmp     RstrFrmDlg
; ============================================================
;  Dispatch routine for "mini-close" icon in vwTextDB
; ============================================================
doMniCls:       lda     #CLOSE
                sta     sysDBData
                jmp     RstrFrmDlg
```

```
; ============================================================
;  Dispatch routine for bottom button in scrollbar (for text).
; ============================================================
txtBot:         ldy     pageNdx
                iny
                cpy     numItems        ;i.e. number of pages
                bcc     10$
                jsr     beep
                rts
10$             ldy     numItems
                dey
                sty     pageNdx
                sty     topItem         ;for shared code
                jsr     reText
                rts
; ============================================================
;  Dispatch routine for top button in scrollbar (for text).
; ============================================================
txtTop:         lda     pageNdx
                bne     10$
                jsr     beep
                rts
10$             lda     #0
                sta     pageNdx
                sta     topItem         ;for shared code
                jsr     reText
                rts
; ============================================================
;  Page down routine for text (called through otherPressVector).
; ============================================================
txtPgDn:        ldy     pageNdx
                iny
                cpy     numItems        ;i.e. number of pages
                bcc     10$
                jsr     beep
                rts
10$             inc     pageNdx
                MoveB   pageNdx,topItem  ;for shared code
                jsr     reText
                rts
; ============================================================
;  Page up routine for text (called through otherPressVector).
; ============================================================

txtPgUp:        lda     pageNdx
                bne     10$
                jsr     beep
                rts
10$             dec     pageNdx
                MoveB   pageNdx,topItem  ;for shared code
                jsr     reText
                rts
```

```
; ============================================================
;  Clear and redraw text area.
; ============================================================
reText:         lda     #0              ;clear
                jsr     SetPattern
                LoadB   r2L,7
                LoadB   r2H,184
                LoadW   r3,5
                LoadW   r4,307
                jsr     Rectangle       ;clear text area
                MoveW   fontLoad,r0
                jsr     LoadCharSet
                jsr     viewText
                jsr     UseSystemFont
                rts
```