

!A

*** End of Pass 1

*** End of Pass 2

```
0800      1          ttl "String Sort Source Code, SS.L"
0800      2      ;
0800      3      ;
0800      4      ; SS.L
0800      5      ;
0800      6      ;
0800      7      ; String Sort Source Code
0800      8      ;
0800      9      ; 2024 February 14
0800     10      ;
0800     11      ;
0800     12      ; DOS 4.5, Build 06
0800     13      ;
0800     14      ; 2024 February 14
0800     15      ;
0800     16      ;
0800     17      ; Start of Source Code: 0x4000
0800     18      ; Start of Symbol List: 0x7800
0800     19      ;
0800     20      ;
0800     21      ; Copyright (c) 2024 February 14 by
0800     22      ; Walland Philip Vrbancic Jr
0800     23      ;
0800     24      ; 6223 East Peabody Street
0800     25      ; Long Beach, California 90808
0800     26      ; Unitied States of America
0800     27      ;
0800     28      ; All Rights Reserved
0800     29      ;
0800     30      ; This software is the confidential and
0800     31      ; proprietary intellectual property of
0800     32      ; Walland Philip Vrbancic Jr
0800     33      ;
0800     34      ;
0800     35      ; CALL SS,F$(A%),I%(A%),A%,B%[,O%]
0800     36      ;
0800     37      ;     SS = Entry address of routine
0800     38      ;
0800     39      ;     F$ = String array to be sorted
0800     40      ;
0800     41      ;         Required in command line
0800     42      ;
0800     43      ;     I% = Returned sorted index array
0800     44      ;
0800     45      ;         Required in command line
0800     46      ;
0800     47      ;     A% = First string array index
0800     48      ;
0800     49      ;         Required in command line
0800     50      ;
0800     51      ;     B% = Last string array index
0800     52      ;
0800     53      ;         Required in command line
0800     54      ;
0800     55      ;     O% = Order of sort, 0:A-Z, -1:Z-A
0800     56      ;
0800     57      ;         Optional in command line
0800     58      ;
0800     59      ; This code will execute at any address.
0800     60      ;
```

```

0800          61 ;
00B7          62 CHRGOT    equ  $B7
0800          63 ;
00EB          64 IFLAG     epz  $EB
00EC          65 F$PTR1    epz  $EC
00EE          66 F$PTR2    epz  $EE
0800          67 ;
00F6          68 VAR%ADR    epz  $F6
00F6          69 I%ADR      epz  $F6
00FA          70 STRPTR1    epz  $FA
00FC          71 STRPTR2    epz  $FC
0800          72 ;
0800          73          enz
0800          74 ;
0000          75 ZERO      equ  $00
0800          76 ;
0110          77 F$SAV     equ  $110
0112          78 I%SAV     equ  $112
0800          79 ;
0114          80 A%VAL     equ  $114
0116          81 B%VAL     equ  $116
0800          82 ;
0118          83 INDEX1    equ  $118
011A          84 INDEX2    equ  $11A
0800          85 ;
011C          86 STRLEN0    equ  $11C
011D          87 STRLEN1    equ  $11D
011E          88 STRLEN2    equ  $11E
0800          89 ;
DEBE          90 CHKCOM     equ  $DEBE
DFE3          91 PTRGET     equ  $DFE3
0800          92 ;
0800          93 ;
0900          94          org  $900
0900          95          obj  $900
0900          96          usr
0900          97 ;
0900          98 ;
0900          99 ; Get F$(A%) string array address.
0900         100 ;
0900 20 BE DE 101          jsr  CHKCOM
0903 20 E3 DF 102          jsr  PTRGET
0906         103 ;
0906 8D 10 01 104          sta  F$SAV
0909 8C 11 01 105          sty  F$SAV+1
090C         106 ;
090C 85 EC    107          sta  F$PTR1
090E 84 ED    108          sty  F$PTR1+1
0910         109 ;
0910         110 ;
0910         111 ; Get I$(A%) variable array address.
0910         112 ;
0910 20 BE DE 113          jsr  CHKCOM
0913 20 E3 DF 114          jsr  PTRGET
0916         115 ;
0916 8D 12 01 116          sta  I%SAV
0919 8C 13 01 117          sty  I%SAV+1
091C         118 ;
091C         119 ;
091C         120 ; Get A% variable address and value.
091C         121 ;

```

```

091C 20 BE DE 122      jsr CHKCOM
091F 20 E3 DF 123      jsr PTRGET
0922      124      ;
0922 85 F6 125      sta VAR%ADR
0924 84 F7 126      sty VAR%ADR+1
0926      127      ;
0926 A0 01 128      ldy #1
0928      129      ;
0928 B1 F6 130      lda (VAR%ADR),Y
092A      131      ;
092A 8D 14 01 132      sta A%VAL
092D 8D 18 01 133      sta INDEX1
0930      134      ;
0930 88 135      dey
0931      136      ;
0931 B1 F6 137      lda (VAR%ADR),Y
0933      138      ;
0933 8D 15 01 139      sta A%VAL+1
0936 8D 19 01 140      sta INDEX1+1
0939      141      ;
0939      142      ;
0939      143      ; Get B% variable address and value.
0939      144      ;
0939 20 BE DE 145      jsr CHKCOM
093C 20 E3 DF 146      jsr PTRGET
093F      147      ;
093F 85 F6 148      sta VAR%ADR
0941 84 F7 149      sty VAR%ADR+1
0943      150      ;
0943 A0 01 151      ldy #1
0945      152      ;
0945 B1 F6 153      lda (VAR%ADR),Y
0947 8D 16 01 154      sta B%VAL
094A      155      ;
094A 88 156      dey
094B      157      ;
094B B1 F6 158      lda (VAR%ADR),Y
094D 8D 17 01 159      sta B%VAL+1
0950      160      ;
0950      161      ;
0950      162      ; Get O% variable address if available and set IFLAG.
0950      163      ;
0950 A9 00 164      lda #ZERO
0952 85 EB 165      sta IFLAG
0954      166      ;
0954 20 B7 00 167      jsr CHRGOT
0957 F0 12 168      beq LOOP1
0959      169      ;
0959 20 BE DE 170      jsr CHKCOM
095C F0 0D 171      beq LOOP1
095E      172      ;
095E 20 E3 DF 173      jsr PTRGET
0961      174      ;
0961 85 F6 175      sta VAR%ADR
0963 84 F7 176      sty VAR%ADR+1
0965      177      ;
0965 A0 01 178      ldy #1
0967      179      ;
0967 B1 F6 180      lda (VAR%ADR),Y
0969 85 EB 181      sta IFLAG
096B      182      ;

```

```

096B      183 ;
096B AD 10 01 184 LOOP1    lda F$SAV
096E 85 EE    185          sta F$PTR2
0970      186 ;
0970 AD 11 01 187          lda F$SAV+1
0973 85 EF    188          sta F$PTR2+1
0975      189 ;
0975 AD 14 01 190          lda A%VAL
0978 8D 1A 01 191          sta INDEX2
097B      192 ;
097B AD 15 01 193          lda A%VAL+1
097E 8D 1B 01 194          sta INDEX2+1
0981      195 ;
0981 A0 00    196          ldy #ZERO
0983      197 ;
0983 84 F6    198          sty I%ADR
0985 84 F7    199          sty I%ADR+1
0987      200 ;
0987 B1 EC    201          lda (F$PTR1),Y
0989 8D 1D 01 202          sta STRLEN1
098C      203 ;
098C C8       204          iny
098D      205 ;
098D B1 EC    206          lda (F$PTR1),Y
098F 85 FA    207          sta STRPTR1
0991      208 ;
0991 C8       209          iny
0992      210 ;
0992 B1 EC    211          lda (F$PTR1),Y
0994 85 FB    212          sta STRPTR1+1
0996      213 ;
0996      214 ;
0996      215 ; Sort the string array.
0996      216 ;
0996 A0 00    217 LOOP2    ldy #ZERO
0998      218 ;
0998 B1 EE    219          lda (F$PTR2),Y
099A 8D 1E 01 220          sta STRLEN2
099D      221 ;
099D C8       222          iny
099E      223 ;
099E B1 EE    224          lda (F$PTR2),Y
09A0 85 FC    225          sta STRPTR2
09A2      226 ;
09A2 C8       227          iny
09A3      228 ;
09A3 B1 EE    229          lda (F$PTR2),Y
09A5 85 FD    230          sta STRPTR2+1
09A7      231 ;
09A7 AD 1E 01 232          lda STRLEN2
09AA CD 1D 01 233          cmp STRLEN1
09AD 90 03    234          bcc >1
09AF      235 ;
09AF AD 1D 01 236          lda STRLEN1
09B2      237 ;
09B2 8D 1C 01 238 ^1      sta STRLEN0
09B5      239 ;
09B5 A0 00    240          ldy #ZERO
09B7      241 ;
09B7 B1 FC    242 ^2      lda (STRPTR2),Y
09B9 D1 FA    243          cmp (STRPTR1),Y

```

```

09BB D0 22      244      bne >1
09BD           245      ;
09BD C8         246      iny
09BE           247      ;
09BE CC 1C 01   248      cpy STRLEN0
09C1 D0 F4      249      bne <2
09C3           250      ;
09C3 AD 1E 01   251      lda STRLEN2
09C6 CD 1D 01   252      cmp STRLEN1
09C9 D0 14      253      bne >1
09CB           254      ;
09CB AD 1B 01   255      lda INDEX2+1
09CE CD 19 01   256      cmp INDEX1+1
09D1 D0 0C      257      bne >1
09D3           258      ;
09D3 AD 1A 01   259      lda INDEX2
09D6 CD 18 01   260      cmp INDEX1
09D9 D0 04      261      bne >1
09DB           262      ;
09DB F0 11      263      beq >4                ; always taken
09DD           264      ;
09DD 90 8C      265      BCCLOC bcc LOOP1
09DF           266      ;
09DF 24 EB      267      ^1 bit IFLAG
09E1 30 03      268      bmi >2
09E3           269      ;
09E3 90 03      270      bcc >3
09E5           271      ;
09E5 18         272      clc
09E6           273      ;
09E6 90 06      274      ^2 bcc >4
09E8           275      ;
09E8 E6 F6      276      ^3 inc I%ADR
09EA D0 02      277      bne >4
09EC           278      ;
09EC E6 F7      279      inc I%ADR+1
09EE           280      ;
09EE EE 1A 01   281      ^4 inc INDEX2
09F1 D0 03      282      bne >5
09F3           283      ;
09F3 EE 1B 01   284      inc INDEX2+1
09F6           285      ;
09F6 38         286      ^5 sec
09F7           287      ;
09F7 AD 16 01   288      lda B%VAL
09FA ED 1A 01   289      sbc INDEX2
09FD           290      ;
09FD AD 17 01   291      lda B%VAL+1
0A00 ED 1B 01   292      sbc INDEX2+1
0A03 90 0C      293      bcc >6
0A05           294      ;
0A05 A5 EE      295      lda F$PTR2
0A07 69 02      296      adc #2
0A09 85 EE      297      sta F$PTR2
0A0B           298      ;
0A0B 90 89      299      bcc LOOP2
0A0D           300      ;
0A0D E6 EF      301      inc F$PTR2+1
0A0F           302      ;
0A0F B0 85      303      bcs LOOP2                ; always taken
0A11           304      ;

```

```

0A11 06 F6      305  ^6      asl I%ADR
0A13 26 F7      306      rol I%ADR+1
0A15           307      ;
0A15 18         308      clc
0A16           309      ;
0A16 A5 F6      310      lda I%ADR
0A18 6D 12 01   311      adc I%SAV
0A1B 85 F6      312      sta I%ADR
0A1D           313      ;
0A1D A5 F7      314      lda I%ADR+1
0A1F 6D 13 01   315      adc I%SAV+1
0A22 85 F7      316      sta I%ADR+1
0A24           317      ;
0A24 A0 01      318      ldy #1
0A26           319      ;
0A26 AD 18 01   320      lda INDEX1
0A29 91 F6      321      sta (I%ADR),Y
0A2B           322      ;
0A2B 88         323      dey
0A2C           324      ;
0A2C AD 19 01   325      lda INDEX1+1
0A2F 91 F6      326      sta (I%ADR),Y
0A31           327      ;
0A31 EE 18 01   328      inc INDEX1
0A34 D0 03      329      bne >7
0A36           330      ;
0A36 EE 19 01   331      inc INDEX1+1
0A39           332      ;
0A39 38         333      ^7      sec
0A3A           334      ;
0A3A AD 16 01   335      lda B%VAL
0A3D ED 18 01   336      sbc INDEX1
0A40           337      ;
0A40 AD 17 01   338      lda B%VAL+1
0A43 ED 19 01   339      sbc INDEX1+1
0A46 90 0D      340      bcc >8
0A48           341      ;
0A48 A5 EC      342      lda F$PTR1
0A4A 69 02      343      adc #2
0A4C 85 EC      344      sta F$PTR1
0A4E           345      ;
0A4E 90 8D      346      bcc BCCLOC
0A50           347      ;
0A50 E6 ED      348      inc F$PTR1+1
0A52           349      ;
0A52 18         350      clc
0A53 90 88      351      bcc BCCLOC      ; always taken
0A55           352      ;
0A55 60         353      ^8      rts
0A56           354      ;
0A56           355      ;

```

BSAVE SS,A\$0900,B,L\$0156

```

0A56           356      usr SS
0A56           357      ;
0A56           358      ;
0A56           359      stt "SS Symbol Table"
0A56           360      ;
0A56           361      ;
0A56           362      end 111

```

*** End of Assembly

Symbol List starts at 0x7800, ends at 0x78FA, used 0x00FA, remaining 0x3E8E

Symbols unsorted:

IFLAG	00EB	F\$PTR1	00EC	F\$PTR2	00EE	VAR%ADR	00F6	I%ADR	00F6
STRPTR1	00FA	STRPTR2	00FC	CHRGOT	00B7	ZERO	0000	F\$SAV	0110
I\$SAV	0112	A%VAL	0114	B%VAL	0116	INDEX1	0118	INDEX2	011A
STRLEN0	011C	STRLEN1	011D	STRLEN2	011E	CHKCOM	DEBE	PTRGET	DFE3
LOOP1	096B	LOOP2	0996	BCCLOC	09DD				

Symbols alphabetically sorted:

A%VAL	0114	B%VAL	0116	BCCLOC	09DD	CHKCOM	DEBE	CHRGOT	00B7
F\$PTR1	00EC	F\$PTR2	00EE	F\$SAV	0110	I%ADR	00F6	I\$SAV	0112
IFLAG	00EB	INDEX1	0118	INDEX2	011A	LOOP1	096B	LOOP2	0996
PTRGET	DFE3	STRLEN0	011C	STRLEN1	011D	STRLEN2	011E	STRPTR1	00FA
STRPTR2	00FC	VAR%ADR	00F6	ZERO	0000				

Symbols numerically sorted:

ZERO	0000	CHRGOT	00B7	IFLAG	00EB	F\$PTR1	00EC	F\$PTR2	00EE
VAR%ADR	00F6	I%ADR	00F6	STRPTR1	00FA	STRPTR2	00FC	F\$SAV	0110
I\$SAV	0112	A%VAL	0114	B%VAL	0116	INDEX1	0118	INDEX2	011A
STRLEN0	011C	STRLEN1	011D	STRLEN2	011E	LOOP1	096B	LOOP2	0996
BCCLOC	09DD	CHKCOM	DEBE	PTRGET	DFE3				