```
RRRRRRRRRRR      UUU          UUU  NNN         NNN     000000000     FFFFFFFFFFFFFFF   FFFFFFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN         NNN     000000000     FFFFFFFFFFFFFFF   FFFFFFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN         NNN     000000000     FFFFFFFFFFFFFFF   FFFFFFFFFFFFFFF
RRR          RRR UUU          UUU  NNN         NNN  000          000 FFF               FFF
RRR          RRR UUU          UUU  NNN         NNN  000          000 FFF               FFF
RRR          RRR UUU          UUU  NNN         NNN  000          000 FFF               FFF
RRR          RRR UUU          UUU  NNNNNN      NNN  000          000 FFF               FFF
RRR          RRR UUU          UUU  NNNNNN      NNN  000          000 FFF               FFF
RRR          RRR UUU          UUU  NNNNNN      NNN  000          000 FFF               FFF
RRRRRRRRRRR      UUU          UUU  NNN   NNN   NNN  000          000 FFFFFFFFFFF       FFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN   NNN   NNN  000          000 FFFFFFFFFFF       FFFFFFFFFFF
RRRRRRRRRRRR     UUU          UUU  NNN   NNN   NNN  000          000 FFFFFFFFFFF       FFFFFFFFFFF
RRR   RRR        UUU          UUU  NNN       NNNNNN 000          000 FFF               FFF
RRR   RRR        UUU          UUU  NNN       NNNNNN 000          000 FFF               FFF
RRR   RRR        UUU          UUU  NNN       NNNNNN 000          000 FFF               FFF
RRR      RRR     UUU          UUU  NNN          NNN 000          000 FFF               FFF
RRR      RRR     UUU          UUU  NNN          NNN 000          000 FFF               FFF
RRR      RRR     UUU          UUU  NNN          NNN 000          000 FFF               FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN          NNN    000000000     FFF               FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN          NNN    000000000     FFF               FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN          NNN    000000000     FFF               FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN          NNN    000000000     FFF               FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN          NNN    000000000     FFF               FFF
```

```
_$2

Sym
---
NDX
NDX
NUM
NUM
OPE
OUT
PAC
PAC
PAC
PAC
PAC
PAC
PAC
PAD
PAG
PAG
PAG
PAG
PAG
PAG
PAG
PER
PUT
RCO
RIN

RLI
RNO
RNO
RTY
SAV
STR
STR
STR
STR
STR
STR

STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
```

```
NN       NN  DDDDDDDD    XX       XX  TTTTTTTTTT  EEEEEEEEEE  XX       XX
NN       NN  DDDDDDDD    XX       XX  TTTTTTTTTT  EEEEEEEEEE  XX       XX
NN       NN  DD       DD  XX       XX      TT      EE          XX       XX
NN       NN  DD       DD  XX       XX      TT      EE          XX       XX
NNNN     NN  DD       DD    XX   XX        TT      EE            XX   XX
NNNN     NN  DD       DD    XX   XX        TT      EE            XX   XX
NN  NN   NN  DD       DD      XX          TT      EEEEEEEE        XX
NN  NN   NN  DD       DD      XX          TT      EEEEEEEE        XX
NN     NNNN  DD       DD    XX   XX        TT      EE            XX   XX
NN     NNNN  DD       DD    XX   XX        TT      EE            XX   XX
NN       NN  DD       DD  XX       XX      TT      EE          XX       XX      ....
NN       NN  DD       DD  XX       XX      TT      EE          XX       XX      ....
NN       NN  DDDDDDDD    XX       XX      TT      EEEEEEEEEE  XX       XX      ....
NN       NN  DDDDDDDD    XX       XX      TT      EEEEEEEEEE  XX       XX      ....

LL             IIIIII      SSSSSSSS
LL             IIIIII      SSSSSSSS
LL               II      SS
LL               II      SS
LL               II      SS
LL               II        SSSSSS
LL               II        SSSSSS
LL               II              SS
LL               II              SS
LL               II              SS
LL               II              SS
LLLLLLLLLL     IIIIII    SSSSSSSS
LLLLLLLLLL     IIIIII    SSSSSSSS
```

```
;     1          0001  0 %TITLE 'NDXTEX -- RUNOFF to TEX conversion'
;     2          0002  0 MODULE NDXTEX (IDENT = 'V04-000'
;     3          0003  0                %BLISS32 [, ADDRESSING_MODE (EXTERNAL = LONG_RELATIVE, NONEXTERNAL = LONG_RELATIVE)]
;     4          0004  0                ) =
;     5          0005  1 BEGIN
;     6          0006  1
;     7          0007  1 !
;     8          0008  1 !*****************************************************************************
;     9          0009  1 !*                                                                           *
;    10          0010  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;    11          0011  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;    12          0012  1 !*  ALL RIGHTS RESERVED.                                                      *
;    13          0013  1 !*                                                                           *
;    14          0014  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED     *
;    15          0015  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
;    16          0016  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER     *
;    17          0017  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY     *
;    18          0018  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY     *
;    19          0019  1 !*  TRANSFERRED.                                                              *
;    20          0020  1 !*                                                                           *
;    21          0021  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE     *
;    22          0022  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT     *
;    23          0023  1 !*  CORPORATION.                                                              *
;    24          0024  1 !*                                                                           *
;    25          0025  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS     *
;    26          0026  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                   *
;    27          0027  1 !*                                                                           *
;    28          0028  1 !*                                                                           *
;    29          0029  1 !*****************************************************************************
;    30          0030  1 !
;    31          0031  1
;    32          0032  1 !++
;    33          0033  1 ! FACILITY:
;    34          0034  1 !   DSR (Digital Standard RUNOFF) /DSRPLUS DSRINDEX/INDEX Utility
;    35          0035  1 !
;    36          0036  1 ! ABSTRACT:
;    37          0037  1 !   This module contains code to convert RUNOFF input text lines to
;    38          0038  1 !   TEX text lines.
;    39          0039  1 !
;    40          0040  1 ! ENVIRONMENT:   Transportable
;    41          0041  1 !
;    42          0042  1 ! AUTHOR:        JPK
;    43          0043  1 !
;    44          0044  1 ! CREATION DATE: March 1983
;    45          0045  1 !
;    46          0046  1 ! MODIFIED BY:
;    47          0047  1 !
;    48          0048  1 !--
```

```
;    50          0049  1 !
;    51          0050  1 ! TABLE OF CONTENTS:
;    52          0051  1 !
;    53          0052  1
;    54          0053  1 FORWARD ROUTINE
;    55          0054  1     RNOTEX : NOVALUE;                   ! Generate TEX output
;    56          0055  1
;    57          0056  1 !
;    58          0057  1 ! INCLUDE FILES:
;    59          0058  1 !
;    60          0059  1
;    61          0060  1 LIBRARY 'NXPORT:XPORT';
;    62          0061  1
;    63        L 0062  1 %IF %BLISS (BLISS32)
;    64          0063  1 %THEN
;    65          0064  1
;    66          0065  1 REQUIRE 'REQ:NDXVMSREQ';
;    67          0346  1
;    68          0347  1 %FI
;    69          0348  1
;    70          0349  1 !
;    71          0350  1 ! MACROS:
;    72          0351  1 !
;    73          0352  1 !
;    74          0353  1 ! EQUATED SYMBOLS:
;    75          0354  1 !
;    76          0355  1
;    77          0356  1 LITERAL
;    78          0357  1     TRUE = 1,
;    79          0358  1     FALSE = 0,
;    80          0359  1     BUFFER_SIZE = 2000;
;    81          0360  1
;    82          0361  1 LITERAL                                 ! Names for various fonts.
;    83          0362  1     NORMAL_FONT = 0,                    ! NOTE: code in RNOTES makes use of the
;    84          0363  1     BOLD_FONT   = 1,                    ! fact that NORMAL_FONT is zero and that
;    85          0364  1     ITALIC_FONT = 2,                    ! (BOLD_FONT OR ITALIC_FONT) is equal to
;    86          0365  1     BOLD_ITALIC = 3;                    ! BOLD_ITALIC.
;    87          0366  1
;    88          0367  1 !
;    89          0368  1 ! OWN STORAGE:
;    90          0369  1 !
;    91          0370  1 !
;    92          0371  1 ! EXTERNAL REFERENCES:
;    93          0372  1 !
;    94          0373  1 EXTERNAL LITERAL
;    95          0374  1     TAB : UNSIGNED (8);
```

```
;    97          0375  1 %SBTTL 'RNOTEX - Generate TEX Output from RUNOFF text'
;    98          0376  1 GLOBAL ROUTINE RNOTEX (RNO_LEN, RNO_PTR, DSC) : NOVALUE =
;    99          0377  1 !++
;   100          0378  1 !
;   101          0379  1 ! FUNCTIONAL DESCRIPTION:
;   102          0380  1 !
;   103          0381  1 !       This routine converts RUNOFF input text to TEX input text
;   104          0382  1 !
;   105          0383  1 ! FORMAL PARAMETERS:
;   106          0384  1 !
;   107          0385  1 !       RNO_LEN - Length of input text
;   108          0386  1 !       RNO_PTR - CH$PTR to input text
;   109          0387  1 !       DSC     - Address of output string descriptor
;   110          0388  1 !
;   111          0389  1 ! IMPLICIT INPUTS:
;   112          0390  1 !
;   113          0391  1 !       None
;   114          0392  1 !
;   115          0393  1 ! IMPLICIT OUTPUTS:
;   116          0394  1 !
;   117          0395  1 !       None
;   118          0396  1 !
;   119          0397  1 ! ROUTINE VALUE:
;   120          0398  1 ! COMPLETION CODES:
;   121          0399  1 !
;   122          0400  1 !       None
;   123          0401  1 !
;   124          0402  1 ! SIDE EFFECTS:
;   125          0403  1 !
;   126          0404  1 !       None
;   127          0405  1 !--
;   128          0406  2     BEGIN
;   129          0407  2
;   130          0408  2     LOCAL
;   131          0409  2         LINE : VECTOR [CH$ALLOCATION (BUFFER_SIZE + 50)],
;   132          0410  2         LEADING_BLANKS,
;   133          0411  2         I_PTR,                          ! Copy of input line pointer
;   134          0412  2         I_LEN,                          ! Copy of input line length
;   135          0413  2         L_PTR,                          ! Pointer to output line
;   136          0414  2         L_LEN,                          ! Length of output line
;   137          0415  2         CURRENT_FONT,                   ! Font currently in use
;   138          0416  2         NEXT_FONT,                      ! Font of next character
;   139          0417  2         CH;                             ! Current character
;   140          0418  2
;   141          0419  2     L_LEN = 0;                          ! No characters in output string
;   142          0420  2     L_PTR = CH$PTR (LINE);              ! Point to beginning of output string
;   143          0421  2     CURRENT_FONT = NORMAL_FONT;         ! Font currently in use
;   144          0422  2     NEXT_FONT = NORMAL_FONT;            ! Font of next character
;   145          0423  2
;   146          0424  2     I_LEN = .RNO_LEN;
;   147          0425  2     I_PTR = CH$PLUS (.RNO_PTR, .RNO_LEN - 1);
;   148          0426  2
;   149          0427  2     DECR I FROM .RNO_LEN - 1 TO 0 DO
;   150          0428  3         BEGIN
;   151          0429  3         !
;   152          0430  3         ! Remove trailing whitespace
;   153          0431  3         !
```

```
;   154     0432  3          CH = CH$RCHAR (.I_PTR);
;   155     0433  3
;   156     0434  3          IF (.CH NEQ %C' ') AND (.CH NEQ TAB) THEN EXITLOOP;
;   157     0435  3
;   158     0436  3          I_LEN = .I;
;   159     0437  3          I_PTR = CH$PLUS (.I_PTR, -1);
;   160     0438  2          END;
;   161     0439  2
;   162     0440  2      I_PTR = .RNO_PTR;
;   163     0441  2      LEADING_BLANKS = 0;
;   164     0442  2
;   165     0443  2      DECR I FROM .I_LEN TO 1 DO
;   166     0444  2          !
;   167     0445  2          ! Count the number of leading blanks
;   168     0446  2          !
;   169     0447  2          IF CH$RCHAR_A (I_PTR) EQL %C' '
;   170     0448  2          THEN
;   171     0449  2              LEADING_BLANKS = .LEADING_BLANKS + 1
;   172     0450  2          ELSE
;   173     0451  3              BEGIN
;   174     0452  3              I_LEN = .I;
;   175     0453  3              I_PTR = CH$PLUS (.I_PTR, -1);       ! Back up to account for overshoot
;   176     0454  3              EXITLOOP;
;   177     0455  2              END;
;   178     0456  2
;   179     0457  2      IF .LEADING_BLANKS
;   180     0458  2      THEN
;   181     0459  3          BEGIN
;   182     0460  3          !
;   183     0461  3          ! Odd number of leading blanks
;   184     0462  3          !
;   185     0463  3          I_LEN = .I_LEN + 1;
;   186     0464  3          I_PTR = CH$PLUS (.I_PTR, -1);
;   187     0465  3
;   188     0466  3          LEADING_BLANKS = .LEADING_BLANKS - 1;
;   189     0467  2          END;
;   190     0468  2
;   191     0469  2      INCR I FROM 1 TO .LEADING_BLANKS / 2 DO
;   192     0470  3          BEGIN
;   193     0471  3          !
;   194     0472  3          ! Insert indent sequence for each pair of leading blanks
;   195     0473  3          !
;   196     0474  3          CH$MOVE (8, CH$PTR (UPLIT ('\mspace ')), .L_PTR);
;   197     0475  3          L_PTR = CH$PLUS (.L_PTR, 8);
;   198     0476  3          L_LEN = .L_LEN + 8;
;   199     0477  2          END;
;   200     0478  2
;   201     0479  2      !
;   202     0480  2      ! Process text portion of line
;   203     0481  2      !
;   204     0482  2      INCR I FROM 1 TO .I_LEN DO
;   205     0483  3          BEGIN
;   206     0484  3          IF .L_LEN GEQ BUFFER_SIZE
;   207     0485  3          THEN
;   208     0486  4              BEGIN
;   209     0487  4              !
;   210     0488  4              ! String too long for buffer
```

```
;  211          0489  4                !
;  212        L 0490  4 %IF %BLISS (BLISS32)
;  213          0491  4 %THEN                                          ! Signal errors in BLISS32
;  214          0492  4
;  215          0493  4                SIGNAL (INDEX$_TRUNCATED, 0, INDEX$_TEXTD, 2, .L_LEN, CH$PTR (LINE));
;  216          0494  4
;  217        U 0495  4 %ELSE                                          ! Use $XPO_PUT_MSG otherwise
;  218        U 0496  4
;  219        U 0497  4                $XPO_PUT_MSG (SEVERITY = WARNING,
;  220        U 0498  4                    STRING = 'string too long - truncated',
;  221        U 0499  4                    STRING = $STR_CONCAT ('entry text: ', (.L_LEN, CH$PTR (LINE))));
;  222        U 0500  4
;  223          0501  4 %FI
;  224          0502  4
;  225          0503  4                EXITLOOP;
;  226          0504  3                END;
;  227          0505  3
;  228          0506  3            CH = CH$RCHAR_A (I_PTR);              ! Get next character
;  229          0507  3
;  230          0508  3            SELECTONE .CH OF
;  231          0509  3                SET
;  232          0510  3
;  233          0511  3                [%C'*']:
;  234          0512  3                    !
;  235          0513  3                    ! Bold next character
;  236          0514  3                    !
;  237          0515  3                    NEXT_FONT = .NEXT_FONT OR BOLD_FONT;
;  238          0516  3
;  239          0517  3                [%C'&']:
;  240          0518  3                    !
;  241          0519  3                    ! Next character is italic
;  242          0520  3                    !
;  243          0521  3                    NEXT_FONT = .NEXT_FONT OR ITALIC_FONT;
;  244          0522  3
;  245          0523  3                [%C'%']:
;  246          0524  4                    BEGIN
;  247          0525  4                    !
;  248          0526  4                    ! Overstrike previous character - ignored
;  249          0527  4                    !
;  250        L 0528  4 %IF %BLISS (BLISS32)
;  251          0529  4 %THEN                                          ! Signal errors for BLISS32
;  252          0530  4
;  253          0531  4                    SIGNAL (INDEX$_OVERSTRK, 0, INDEX$_TEXTD, 2, .RNO_LEN, .RNO_PTR);
;  254          0532  4
;  255        U 0533  4 %ELSE                                          ! Use $XPO_PUT_MSG otherwise
;  256        U 0534  4
;  257        U 0535  4                    $XPO_PUT_MSG (SEVERITY = WARNING,
;  258        U 0536  4                        STRING = 'The following line contains an overstrike sequence',
;  259        U 0537  4                        STRING = $STR_CONCAT ('entry text: ', (.RNO_LEN, .RNO_PTR)));
;  260        U 0538  4
;  261          0539  4 %FI
;  262          0540  4
;  263          0541  4                    CH$RCHAR_A (I_PTR);           ! Ignore next character
;  264          0542  4                    I = .I + 1;
;  265          0543  3                    END;
;  266          0544  3
;  267          0545  3                [OTHERWISE]:
```

```
;  268     0546  4                    BEGIN
;  269     0547  4                    !
;  270     0548  4                    ! A 'normal' character
;  271     0549  4                    !
;  272     0550  4                    IF .CH EQL %C'_'
;  273     0551  4                    THEN
;  274     0552  5                        BEGIN
;  275     0553  5                        !
;  276     0554  5                        ! Character is quoted.
;  277     0555  5                        ! Get character.
;  278     0556  5                        !
;  279     0557  5                        CH = CH$RCHAR_A (I_PTR);
;  280     0558  5                        I = .I + 1;
;  281     0559  4                        END;
;  282     0560  4
;  283     0561  5                    IF  (.CURRENT_FONT NEQ .NEXT_FONT)
;  284     0562  5                    AND (.CH NEQ %C' ')
;  285     0563  4                    THEN
;  286     0564  5                        BEGIN
;  287     0565  5                        !
;  288     0566  5                        ! Next character is not of the same font and is non-blank.
;  289     0567  5                        !
;  290     0568  5                        IF .CURRENT_FONT NEQ NORMAL_FONT
;  291     0569  5                        THEN
;  292     0570  6                            BEGIN
;  293     0571  6                            !
;  294     0572  6                            ! Turn off old font.
;  295     0573  6                            !
;  296     0574  6                            CH$WCHAR_A (%C'}', L_PTR);
;  297     0575  6                            L_LEN = .L_LEN + 1;
;  298     0576  5                            END;
;  299     0577  5
;  300     0578  5                    IF .NEXT_FONT NEQ NORMAL_FONT
;  301     0579  5                    THEN
;  302     0580  6                        BEGIN
;  303     0581  6                        !
;  304     0582  6                        ! Turn on new font.
;  305     0583  6                        !
;  306     0584  6                        CH$WCHAR_A (%C'{', L_PTR);
;  307     0585  6                        CH$WCHAR_A (%C'\', L_PTR);
;  308     0586  6
;  309     0587  6                        SELECTONE .NEXT_FONT OF
;  310     0588  6                            SET
;  311     0589  6
;  312     0590  6                            [BOLD_FONT]:
;  313     0591  7                                BEGIN
;  314     0592  7                                CH$WCHAR_A (%C'b', L_PTR);
;  315     0593  7                                CH$WCHAR_A (%C'f', L_PTR);
;  316     0594  6                                END;
;  317     0595  6
;  318     0596  6                            [ITALIC_FONT]:
;  319     0597  7                                BEGIN
;  320     0598  7                                CH$WCHAR_A (%C'i', L_PTR);
;  321     0599  7                                CH$WCHAR_A (%C't', L_PTR);
;  322     0600  6                                END;
;  323     0601  6
;  324     0602  6                            [BOLD_ITALIC]:
```

```
;   325          0603  7                                BEGIN
;   326          0604  7                                CH$WCHAR_A (%C'b', L_PTR);
;   327          0605  7                                CH$WCHAR_A (%C'i', L_PTR);
;   328          0606  6                                END;
;   329          0607  6
;   330          0608  6                            TES;
;   331          0609  6
;   332          0610  6                        CH$WCHAR_A (%C' ', L_PTR);
;   333          0611  6                        L_LEN = .L_LEN + 5;
;   334          0612  5                        END;
;   335          0613  5
;   336          0614  5                    CURRENT_FONT = .NEXT_FONT;
;   337          0615  4                    END;
;   338          0616  4
;   339          0617  4                NEXT_FONT = NORMAL_FONT;
;   340          0618  4
;   341          0619  5                IF (.CH LSS %C' ') OR (.CH GTR %O'176')
;   342          0620  4                THEN
;   343          0621  5                    BEGIN
;   344          0622  5                    !
;   345          0623  5                    ! Character is a control character which are not allowed
;   346          0624  5                    !
;   347        L 0625  5 %IF %BLISS (BLISS32)
;   348          0626  5 %THEN                                       ! Signal errors in BLISS32
;   349          0627  5
;   350          0628  5                    SIGNAL (INDEX$_CTRLCHAR, 0, INDEX$_TEXTD, 2, .RNO_LEN, .RNO_PTR);
;   351          0629  5
;   352        U 0630  5 %ELSE                                       ! Use $XPO_PUT_MSG otherwise
;   353        U 0631  5
;   354        U 0632  5                    $XPO_PUT_MSG (SEVERITY = WARNING,
;   355        U 0633  5                        STRING = 'the following line contains control characters - ignored',
;   356        U 0634  5                        STRING = $STR_CONCAT ('entry text: ', (.RNO_LEN, .RNO_PTR)));
;   357        U 0635  5
;   358          0636  5 %FI
;   359          0637  5
;   360          0638  5                    END
;   361          0639  4                ELSE
;   362          0640  5                    BEGIN
;   363          0641  5
;   364          0642  5                    SELECTONE .CH OF
;   365          0643  5                        SET
;   366          0644  5
;   367          0645  5                        [%C' ']:
;   368          0646  6                            BEGIN
;   369          0647  6                            CH$MOVE (4, CH$PTR (UPLIT ('{\_}')), .L_PTR);
;   370          0648  6                            L_PTR = CH$PLUS (.L_PTR, 4);
;   371          0649  6                            L_LEN = .L_LEN + 4;
;   372          0650  5                            END;
;   373          0651  5
;   374          0652  5                        [%C'#']:
;   375          0653  6                            BEGIN
;   376          0654  6                            CH$MOVE (4, CH$PTR (UPLIT ('{\#}')), .L_PTR);
;   377          0655  6                            L_PTR = CH$PLUS (.L_PTR, 4);
;   378          0656  6                            L_LEN = .L_LEN + 4;
;   379          0657  5                            END;
;   380          0658  5
;   381          0659  5                        [%C'$']:
```

```
;  382     0660  6                        BEGIN
;  383     0661  6                        CH$MOVE (4, CH$PTR (UPLIT ('{\$}')), .L_PTR);
;  384     0662  6                        L_PTR = CH$PLUS (.L_PTR, 4);
;  385     0663  6                        L_LEN = .L_LEN + 4;
;  386     0664  5                        END;
;  387     0665  5
;  388     0666  5                    [%C'%']:
;  389     0667  6                        BEGIN
;  390     0668  6                        CH$MOVE (4, CH$PTR (UPLIT ('{\%}')), .L_PTR);
;  391     0669  6                        L_PTR = CH$PLUS (.L_PTR, 4);
;  392     0670  6                        L_LEN = .L_LEN + 4;
;  393     0671  5                        END;
;  394     0672  5
;  395     0673  5                    [%C'&']:
;  396     0674  6                        BEGIN
;  397     0675  6                        CH$MOVE (4, CH$PTR (UPLIT ('{\&}')), .L_PTR);
;  398     0676  6                        L_PTR = CH$PLUS (.L_PTR, 4);
;  399     0677  6                        L_LEN = .L_LEN + 4;
;  400     0678  5                        END;
;  401     0679  5
;  402     0680  5                    [%C'\']:
;  403     0681  6                        BEGIN
;  404     0682  6                        CH$MOVE (9, CH$PTR (UPLIT ('{\rslash}')), .L_PTR);
;  405     0683  6                        L_PTR = CH$PLUS (.L_PTR, 9);
;  406     0684  6                        L_LEN = .L_LEN + 9;
;  407     0685  5                        END;
;  408     0686  5
;  409     0687  5                    [%C'@']:
;  410     0688  6                        BEGIN
;  411     0689  6                        CH$MOVE (4, CH$PTR (UPLIT ('{\@}')), .L_PTR);
;  412     0690  6                        L_PTR = CH$PLUS (.L_PTR, 4);
;  413     0691  6                        L_LEN = .L_LEN + 4;
;  414     0692  5                        END;
;  415     0693  5
;  416     0694  5                    [%C'^']:
;  417     0695  6                        BEGIN
;  418     0696  6                        CH$MOVE (4, CH$PTR (UPLIT ('{\^}')), .L_PTR);
;  419     0697  6                        L_PTR = CH$PLUS (.L_PTR, 4);
;  420     0698  6                        L_LEN = .L_LEN + 4;
;  421     0699  5                        END;
;  422     0700  5
;  423     0701  5                    [%C'{']:
;  424     0702  6                        BEGIN
;  425     0703  6                        CH$WCHAR_A (%C'\', L_PTR);
;  426     0704  6                        CH$WCHAR_A (%C'{', L_PTR);
;  427     0705  6                        L_LEN = .L_LEN + 2;
;  428     0706  5                        END;
;  429     0707  5
;  430     0708  5                    [%C'}']:
;  431     0709  6                        BEGIN
;  432     0710  6                        CH$WCHAR_A (%C'\', L_PTR);
;  433     0711  6                        CH$WCHAR_A (%C'}', L_PTR);
;  434     0712  6                        L_LEN = .L_LEN + 2;
;  435     0713  5                        END;
;  436     0714  5
;  437     0715  5                    [OTHERWISE]:
;  438     0716  6                        BEGIN
```

```
;  439          0717  6                                !
;  440          0718  6                                ! A real normal character
;  441          0719  6                                !
;  442          0720  6                                CH$WCHAR_A (.CH, L_PTR);
;  443          0721  6                                L_LEN = .L_LEN + 1;
;  444          0722  5                                END;
;  445          0723  5
;  446          0724  5                           TES;
;  447          0725  5
;  448          0726  4                       END;
;  449          0727  4
;  450          0728  3                   END;
;  451          0729  3
;  452          0730  3               TES;
;  453          0731  3
;  454          0732  2           END;
;  455          0733  2
;  456          0734  2       IF .CURRENT_FONT NEQ NORMAL_FONT
;  457          0735  2       THEN
;  458          0736  3           BEGIN
;  459          0737  3           !
;  460          0738  3           ! Turn off bold and italics
;  461          0739  3           !
;  462          0740  3           CH$WCHAR_A (%C'}', L_PTR);
;  463          0741  3           L_LEN = .L_LEN + 1;
;  464          0742  2           END;
;  465          0743  2
;  466          0744  2       !
;  467          0745  2       ! Copy line to output string
;  468          0746  2       !
;  469          0747  2       $STR_COPY (STRING = (.L_LEN, CH$PTR (LINE)), TARGET = .DSC);
;  470          0748  1       END;


                                                                .TITLE  NDXTEX NDXTEX -- RUNOFF to TEX conversion
                                                                .IDENT  \V04-000\

                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

                                20  65  63  61  70  73  6D  5C  00000 P.AAA:  .ASCII  <92>\mspace \                                ;
                                                7D  5F  5C  7B  00008 P.AAB:  .ASCII  \{\<92>\_}\                                ;
                                                7D  23  5C  7B  0000C P.AAC:  .ASCII  \{\<92>\#}\                                ;
                                                7D  24  5C  7B  00010 P.AAD:  .ASCII  \{\<92>\$}\                                ;
                                                7D  25  5C  7B  00014 P.AAE:  .ASCII  \{\<92>\%}\                                ;
                                                7D  26  5C  7B  00018 P.AAF:  .ASCII  \{\<92>\&}\                                ;
                    00  00  00  7D  68  73  61  6C  73  72  5C  7B  0001C P.AAG:  .ASCII  \{\<92>\rslash}\<0><0><0>              ;
                                                7D  40  5C  7B  00028 P.AAH:  .ASCII  \{\<92>\@}\                                ;
                                                7D  5E  5C  7B  0002C P.AAI:  .ASCII  \{\<92>\^}\                                ;

                                                                .EXTRN  DSRINDEX$_BADLOGIC
                                                                .EXTRN  DSRINDEX$_BADVALUE
                                                                .EXTRN  DSRINDEX$_INSVIRMEM
                                                                .EXTRN  DSRINDEX$_LINELENG
                                                                .EXTRN  DSRINDEX$_NOREF
                                                                .EXTRN  DSRINDEX$_OPENIN
                                                                .EXTRN  DSRINDEX$_OPENOUT
                                                                .EXTRN  DSRINDEX$_TOOMANY
```

```
                                                .EXTRN  DSRINDEX$_VALERR
                                                .EXTRN  DSRINDEX$_CANTBAL
                                                .EXTRN  DSRINDEX$_CLOSEQUOT
                                                .EXTRN  DSRINDEX$_CONFQUAL
                                                .EXTRN  DSRINDEX$_CTRLCHAR
                                                .EXTRN  DSRINDEX$_DOESNTFIT
                                                .EXTRN  DSRINDEX$_DUPBEGIN
                                                .EXTRN  DSRINDEX$_EMPTYIN
                                                .EXTRN  DSRINDEX$_IGNORED
                                                .EXTRN  DSRINDEX$_INVINPUT
                                                .EXTRN  DSRINDEX$_INVRECORD
                                                .EXTRN  DSRINDEX$_LASTCONT
                                                .EXTRN  DSRINDEX$_NOBEGIN
                                                .EXTRN  DSRINDEX$_NOEND
                                                .EXTRN  DSRINDEX$_NOINDEX
                                                .EXTRN  DSRINDEX$_NOLIST
                                                .EXTRN  DSRINDEX$_OVERSTRK
                                                .EXTRN  DSRINDEX$_SKIPPED
                                                .EXTRN  DSRINDEX$_SYNTAX
                                                .EXTRN  DSRINDEX$_TEXFILE
                                                .EXTRN  DSRINDEX$_TOODEEP
                                                .EXTRN  DSRINDEX$_TOOFEW
                                                .EXTRN  DSRINDEX$_TRUNCATED
                                                .EXTRN  DSRINDEX$_COMPLETE
                                                .EXTRN  DSRINDEX$_CREATED
                                                .EXTRN  DSRINDEX$_IDENT
                                                .EXTRN  DSRINDEX$_PROCFILE
                                                .EXTRN  DSRINDEX$_TEXT, DSRINDEX$_TEXTD
                                                .EXTRN  DSRINDEX$_TMS11
                                                .EXTRN  TAB, XST$COPY, STR$FAILURE

                                                .PSECT  $CODE$,NOWRT,2

                                    0FFC 00000  .ENTRY  RNOTEX, Save R2,R3,R4,R5,R6,R7,R8,R9,R10,-  ; 0376
                                                        R11
                          5E    F7EC    CE  9E 00002  MOVAB  -2068(SP), SP
                          58      10    AE  9E 00007  MOVAB  LINE, L_PTR                            ; 0420
                                        7E  D4 0000B  CLRL   CURRENT_FONT                           ; 0421
                                        5A  7C 0000D  CLRQ   NEXT_FONT                              ; 0422
                          57      04    AC  D0 0000F  MOVL   RNO_LEN, I_LEN                         ; 0424
                56        08    AC      04  AC  C1 00013  ADDL3  RNO_LEN, RNO_PTR, R6               ; 0425
                                        56  D7 00019  DECL   I_PTR
                          50      04    AC  D0 0001B  MOVL   RNO_LEN, I                             ; 0427
                                        16  11 0001F  BRB    3$
                          59            66  9A 00021  1$:   MOVZBL (I_PTR), CH                     ; 0432
                          20            59  D1 00024  CMPL   CH, #32                              ; 0434
                                        09  13 00027  BEQL   2$
                000000000G  8F          59  D1 00029  CMPL   CH, #TAB
                                        08  12 00030  BNEQ   4$
                          57            50  D0 00032  2$:   MOVL   I, I_LEN                         ; 0436
                                        56  D7 00035  DECL   I_PTR                                 ; 0437
                          E7            50  F4 00037  3$:   SOBGEQ I, 1$                          ; 0427
                          56      08    AC  D0 0003A  4$:   MOVL   RNO_PTR, I_PTR                   ; 0440
                                        51  D4 0003E  CLRL   LEADING_BLANKS                        ; 0441
                          50      01    A7  9E 00040  MOVAB  1(R7), I                               ; 0443
                                        13  11 00044  BRB    7$
                          52            86  9A 00046  5$:   MOVZBL (I_PTR)+, R2                    ; 0447
```

```
                                20              52  91 00049          CMPB    R2, #32                                 ;
                                                04  12 0004C          BNEQ    6$                                      ;
                                                51  D6 0004E          INCL    LEADING_BLANKS                          ; 0449
                                                07  11 00050          BRB     7$                                      ;
                                57              50  D0 00052 6$:      MOVL    I, I_LEN                                ; 0452
                                                56  D7 00055          DECL    I_PTR                                   ; 0453
                                                03  11 00057          BRB     8$                                      ; 0451
                                EA              50  F5 00059 7$:      SOBGTR  I, 5$                                   ; 0447
                                06              51  E9 0005C 8$:      BLBC    LEADING_BLANKS, 9$                      ; 0457
                                                57  D6 0005F          INCL    I_LEN                                   ; 0463
                                                56  D7 00061          DECL    I_PTR                                   ; 0464
                                                51  D7 00063          DECL    LEADING_BLANKS                          ; 0466
                    08  AE          51             02  C7 00065 9$:      DIVL3   #2, LEADING_BLANKS, 8(SP)               ; 0469
                                            04  AE  D4 0006A          CLRL    I                                       ;
                                                0E  11 0006D          BRB     11$                                     ;
                        68 00000000'    EF          08  28 0006F 10$:     MOVC3   #8, P.AAA, (L_PTR)                      ; 0474
                                        58          08  C0 00077          ADDL2   #8, L_PTR                               ; 0475
                                        5B          08  C0 0007A          ADDL2   #8, L_LEN                               ; 0476
                        EC      04      AE      08  AE  F3 0007D 11$:     AOBLEQ  8(SP), I, 10$                           ; 0469
                                            08  AE  D4 00083          CLRL    I                                       ; 0482
                                                64  11 00086          BRB     16$                                     ;
                            000007D0    8F          5B  D1 00088 12$:     CMPL    L_LEN, #2000                            ; 0484
                                                1F  19 0008F          BLSS    13$                                     ;
                                            14  AE  9F 00091          PUSHAB  LINE                                    ; 0493
                                                5B  DD 00094          PUSHL   L_LEN                                   ;
                                                02  DD 00096          PUSHL   #2                                      ;
                                    00000000G   8F  DD 00098          PUSHL   #DSRINDEX$_TEXTD                        ;
                                                7E  D4 0009E          CLRL    -(SP)                                   ;
                                    00000000G   8F  DD 000A0          PUSHL   #DSRINDEX$_TRUNCATED                    ;
                            00000000G   00          06  FB 000A6          CALLS   #6, LIB$SIGNAL                          ;
                                              0192  31  00 000AD          BRW     42$                                     ; 0486
                                59                86  9A 000B0 13$:     MOVZBL  (I_PTR)+, CH                            ; 0506
                                2A                59  D1 000B3          CMPL    CH, #42                                 ; 0511
                                                  05  12 000B6          BNEQ    14$                                     ;
                                5A                01  88 000B8          BISB2   #1, NEXT_FONT                           ; 0515
                                                  2F  11 000BB          BRB     16$                                     ;
                                26                59  D1 000BD 14$:     CMPL    CH, #38                                 ; 0517
                                                  05  12 000C0          BNEQ    15$                                     ;
                                5A                02  88 000C2          BISB2   #2, NEXT_FONT                           ; 0521
                                                  25  11 000C5          BRB     16$                                     ;
                                25                59  D1 000C7 15$:     CMPL    CH, #37                                 ; 0523
                                                  23  12 000CA          BNEQ    17$                                     ;
                                7E          04    AC  7D 000CC          MOVQ    RNO_LEN, -(SP)                          ; 0531
                                                  02  DD 000D0          PUSHL   #2                                      ;
                                    00000000G     8F  DD 000D2          PUSHL   #DSRINDEX$_TEXTD                        ;
                                                  7E  D4 000D8          CLRL    -(SP)                                   ;
                                    00000000G     8F  DD 000DA          PUSHL   #DSRINDEX$_OVERSTRK                     ;
                    00000000G   00                  06  FB 000E0          CALLS   #6, LIB$SIGNAL                          ;
                                                    56  D6 000E7          INCL    I_PTR                                   ; 0541
                                            08  AE  D6 000E9          INCL    I                                       ; 0542
                                              014C  31  00 000EC 16$:     BRW     41$                                     ; 0508
                    0000005F    8F                  59  D1 000EF 17$:     CMPL    CH, #95                                 ; 0550
                                                    06  12 000F6          BNEQ    18$                                     ;
                                59                  86  9A 000F8          MOVZBL  (I_PTR)+, CH                            ; 0557
                                            08  AE  D6 000FB          INCL    I                                       ; 0558
                                5A                  6E  D1 000FE 18$:     CMPL    CURRENT_FONT, NEXT_FONT                 ; 0561
                                                    4E  13 00101          BEQL    25$                                     ;
```

```
                         20              59  D1 00103          CMPL    CH, #32                                 ; 0562
                                         49  13 00106          BEQL    25$                                     ;
                                         6E  D5 00108          TSTL    CURRENT_FONT                            ; 0568
                                         06  13 0010A          BEQL    19$                                     ;
                         88        7D    8F  90 0010C          MOVB    #125, (L_PTR)+                          ; 0574
                                         5B  D6 00110          INCL    L_LEN                                   ; 0575
                                         5A  D5 00112 19$:     TSTL    NEXT_FONT                               ; 0578
                                         38  13 00114          BEQL    24$                                     ;
                         88      5C7B    8F  B0 00116          MOVW    #23675, (L_PTR)+                        ; 0584
                         01              5A  D1 0011B          CMPL    NEXT_FONT, #1                           ; 0590
                                         0A  12 0011E          BNEQ    20$                                     ;
                         88        62    8F  90 00120          MOVB    #98, (L_PTR)+                           ; 0592
                         68        66    8F  90 00124          MOVB    #102, (L_PTR)                           ; 0593
                                         1C  11 00128          BRB     22$                                     ;
                         02              5A  D1 0012A 20$:     CMPL    NEXT_FONT, #2                           ; 0596
                                         0A  12 0012D          BNEQ    21$                                     ;
                         88        69    8F  90 0012F          MOVB    #105, (L_PTR)+                          ; 0598
                         68        74    8F  90 00133          MOVB    #116, (L_PTR)                           ; 0599
                                         0D  11 00137          BRB     22$                                     ;
                         03              5A  D1 00139 21$:     CMPL    NEXT_FONT, #3                           ; 0602
                                         0A  12 0013C          BNEQ    23$                                     ;
                         88        62    8F  90 0013E          MOVB    #98, (L_PTR)+                           ; 0604
                         68        69    8F  90 00142          MOVB    #105, (L_PTR)                           ; 0605
                                         58  D6 00146 22$:     INCL    L_PTR                                   ;
                         88              20  90 00148 23$:     MOVB    #32, (L_PTR)+                           ; 0610
                         5B              05  C0 0014B          ADDL2   #5, L_LEN                               ; 0611
                         6E              5A  D0 0014E 24$:     MOVL    NEXT_FONT, CURRENT_FONT                 ; 0614
                                         5A  D4 00151 25$:     CLRL    NEXT_FONT                               ; 0617
                         20              59  D1 00153          CMPL    CH, #32                                 ; 0619
                                         09  19 00156          BLSS    26$                                     ;
            0000007E     8F              59  D1 00158          CMPL    CH, #126                                ;
                                         1D  15 0015F          BLEQ    27$                                     ;
                         7E        04    AC  7D 00161 26$:     MOVQ    RNO_LEN, -(SP)                          ; 0628
                                         02  DD 00165          PUSHL   #2                                      ;
                             00000000G   8F  DD 00167          PUSHL   #DSRINDEX$_TEXTD                        ;
                                         7E  D4 0016D          CLRL    -(SP)                                   ;
                             00000000G   8F  DD 0016F          PUSHL   #DSRINDEX$_CTRLCHAR                     ;
            00000000G        00          06  FB 00175          CALLS   #6, LIB$SIGNAL                          ;
                                         61  11 0017C          BRB     33$                                     ; 0619
            0000005F     8F              59  D1 0017E 27$:     CMPL    CH, #95                                 ; 0645
                                         09  12 00185          BNEQ    28$                                     ;
                         68 00000000'    EF  D0 00187          MOVL    P.AAB, (L_PTR)                          ; 0647
                                         73  11 0018E          BRB     36$                                     ; 0648
                         23              59  D1 00190 28$:     CMPL    CH, #35                                 ; 0652
                                         09  12 00193          BNEQ    29$                                     ;
                         68 00000000'    EF  D0 00195          MOVL    P.AAC, (L_PTR)                          ; 0654
                                         65  11 0019C          BRB     36$                                     ; 0655
                         24              59  D1 0019E 29$:     CMPL    CH, #36                                 ; 0659
                                         09  12 001A1          BNEQ    30$                                     ;
                         68 00000000'    EF  D0 001A3          MOVL    P.AAD, (L_PTR)                          ; 0661
                                         57  11 001AA          BRB     36$                                     ; 0662
                         25              59  D1 001AC 30$:     CMPL    CH, #37                                 ; 0666
                                         09  12 001AF          BNEQ    31$                                     ;
                         68 00000000'    EF  D0 001B1          MOVL    P.AAE, (L_PTR)                          ; 0668
                                         49  11 001B8          BRB     36$                                     ; 0669
                         26              59  D1 001BA 31$:     CMPL    CH, #38                                 ; 0673
                                         09  12 001BD          BNEQ    32$                                     ;
```

```
                          68 00000000'   EF   D0 001BF             MOVL    P.AAF, (L_PTR)                          ; 0675
                                         3B   11 001C6             BRB     36$                                     ; 0676
                     0000005C   8F               59   D1 001C8 32$:  CMPL    CH, #92                                 ; 0680
                                                 10   12 001CF        BNEQ    34$                                     ;
                   68 00000000'   EF             09   28 001D1        MOVC3   #9, P.AAG, (L_PTR)                      ; 0682
                                  58             09   C0 001D9        ADDL2   #9, L_PTR                               ; 0683
                                  5B             09   C0 001DC        ADDL2   #9, L_LEN                               ; 0684
                                                 5A   11 001DF 33$:  BRB     41$                                     ; 0642
                     00000040   8F               59   D1 001E1 34$:  CMPL    CH, #64                                 ; 0687
                                                 09   12 001E8        BNEQ    35$                                     ;
                          68 00000000'   EF      D0   001EA          MOVL    P.AAH, (L_PTR)                          ; 0689
                                         10   11 001F1             BRB     36$                                     ; 0690
                     0000005E   8F               59   D1 001F3 35$:  CMPL    CH, #94                                 ; 0694
                                                 0F   12 001FA        BNEQ    37$                                     ;
                          68 00000000'   EF      D0   001FC          MOVL    P.AAI, (L_PTR)                          ; 0696
                                  58             04   C0 00203 36$:  ADDL2   #4, L_PTR                               ; 0697
                                  5B             04   C0 00206        ADDL2   #4, L_LEN                               ; 0698
                                                 30   11 00209        BRB     41$                                     ; 0642
                     0000007B   8F               59   D1 0020B 37$:  CMPL    CH, #123                                ; 0701
                                                 0A   12 00212        BNEQ    38$                                     ;
                                  88       5C    8F   90 00214        MOVB    #92, (L_PTR)+                           ; 0703
                                  68       7B    8F   90 00218        MOVB    #123, (L_PTR)                           ; 0704
                                                 11   11 0021C        BRB     39$                                     ;
                     0000007D   8F               59   D1 0021E 38$:  CMPL    CH, #125                                ; 0708
                                                 0F   12 00225        BNEQ    40$                                     ;
                                  88       5C    8F   90 00227        MOVB    #92, (L_PTR)+                           ; 0710
                                  68       7D    8F   90 0022B        MOVB    #125, (L_PTR)                           ; 0711
                                                 58   D6 0022F 39$:  INCL    L_PTR                                   ;
                                  5B             02   C0 00231        ADDL2   #2, L_LEN                               ; 0712
                                                 05   11 00234        BRB     41$                                     ; 0642
                                  88             59   90 00236 40$:  MOVB    CH, (L_PTR)+                            ; 0720
                                                 5B   D6 00239        INCL    L_LEN                                   ; 0721
            FE46        08   AE                  57   F1 0023B 41$:  ACBL    I_LEN, #1, I, 12$                       ; 0482
                                                 6E   D5 00242 42$:  TSTL    CURRENT_FONT                            ; 0734
                                                 06   13 00244        BEQL    43$                                     ;
                                  88       7D    8F   90 00246        MOVB    #125, (L_PTR)+                          ; 0740
                                                 5B   D6 0024A        INCL    L_LEN                                   ; 0741
                          0C   AE                5B   B0 0024C 43$:  MOVW    L_LEN, $STR$STRING                      ; 0747
                          0E   AE                0E   90 00250        MOVB    #14, $STR$STRING+2                      ;
                          0F   AE                01   90 00254        MOVB    #1, $STR$STRING+3                       ;
                          10   AE          14    AE   9E 00258        MOVAB   LINE, $STR$STRING+4                     ;
                                     00000000G   EF   9F 0025D        PUSHAB  STR$FAILURE                             ;
                                                 7E   D4 00263        CLRL    -(SP)                                   ;
                                           0C    AC   DD 00265        PUSHL   DSC                                     ;
                                           18    AE   9F 00268        PUSHAB  $STR$STRING                             ;
                                                 7E   D4 0026B        CLRL    -(SP)                                   ;
                              00000000G    EF             05   FB 0026D        CALLS   #5, XST$COPY                    ;
                                                          04   00274        RET                                     ; 0748

; Routine Size:  629 bytes,    Routine Base:  $CODE$ + 0000

;  471          0749  1
;  472          0750  1 END                                    ! End of module
;  473          0751  0 ELUDOM
```

```
                                                        .EXTRN  LIB$SIGNAL

;                                   PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  $PLIT$                         48  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                        629  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)


;                                 Library Statistics
;
;                                          -------- Symbols --------      Pages       Processing
;       File                               Total    Loaded   Percent      Mapped      Time
;
;  _$255$DUA28:[SYSLIB]XPORT.L32;1           590        20         3       252          00:00.4


;                                   COMMAND QUALIFIERS
;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:NDXTEX/OBJ=OBJ$:NDXTEX MSRC$:NDXTEX/UPDATE=(ENH$:NDXTEX)

; Size:         629 code + 48 data bytes
; Run Time:        00:16.1
; Elapsed Time:    00:33.5
; Lines/CPU Min:    2795
; Lexemes/CPU-Min: 25704
; Memory Used:  265 pages
; Compilation Complete
```