```
PPPPPPPPPPPP        AAAAAAAAA        SSSSSSSSSSS  RRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
PPPPPPPPPPPP        AAAAAAAAA        SSSSSSSSSSS  RRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
PPPPPPPPPPPP        AAAAAAAAA        SSSSSSSSSSS  RRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
PPP          PPP   AAA         AAA  SSS          RRR         RRR        TTT        LLL
PPP          PPP   AAA         AAA  SSS          RRR         RRR        TTT        LLL
PPP          PPP   AAA         AAA  SSS          RRR         RRR        TTT        LLL
PPP          PPP   AAA         AAA  SSS          RRR         RRR        TTT        LLL
PPP          PPP   AAA         AAA  SSS          RRR         RRR        TTT        LLL
PPP          PPP   AAA         AAA  SSS          RRR         RRR        TTT        LLL
PPPPPPPPPPPP       AAA         AAA     SSSSSSSSS     RRRRRRRRRRR        TTT        LLL
PPPPPPPPPPPP       AAA         AAA     SSSSSSSSS     RRRRRRRRRRR        TTT        LLL
PPPPPPPPPPPP       AAA         AAA     SSSSSSSSS     RRRRRRRRRRR        TTT        LLL
PPP                AAAAAAAAAAAAAAA           SSS   RRR   RRR          TTT        LLL
PPP                AAAAAAAAAAAAAAA           SSS   RRR   RRR          TTT        LLL
PPP                AAAAAAAAAAAAAAA           SSS   RRR   RRR          TTT        LLL
PPP                AAA         AAA           SSS   RRR      RRR       TTT        LLL
PPP                AAA         AAA           SSS   RRR      RRR       TTT        LLL
PPP                AAA         AAA           SSS   RRR      RRR       TTT        LLL
PPP                AAA         AAA  SSSSSSSSSSS   RRR         RRR     TTT        LLLLLLLLLLLLLLL
PPP                AAA         AAA  SSSSSSSSSSS   RRR         RRR     TTT        LLLLLLLLLLLLLLL
PPP                AAA         AAA  SSSSSSSSSSS   RRR         RRR     TTT        LLLLLLLLLLLLLLL
PPP                AAA         AAA  SSSSSSSSSSS   RRR         RRR     TTT        LLLLLLLLLLLLLLL
```

```
PPPPPPPP      AAAAAA      SSSSSSSS  RRRRRRRR    EEEEEEEEEE    AAAAAA    BBBBBBBB      000000      000000
PPPPPPPP      AAAAAA      SSSSSSSS  RRRRRRRR    EEEEEEEEEE    AAAAAA    BBBBBBBB      000000      000000
PP       PP  AA      AA  SS        RR      RR  EE            AA      AA  BB      BB  00      00  00      00
PP       PP  AA      AA  SS        RR      RR  EE            AA      AA  BB      BB  00      00  00      00
PP       PP  AA      AA  SS        RR      RR  EE            AA      AA  BB      BB  00      00  00      00
PP       PP  AA      AA  SS        RR      RR  EE            AA      AA  BB      BB  00      00  00      00
PPPPPPPP     AA      AA    SSSSSS  RRRRRRRR    EEEEEEEE      AA      AA  BBBBBBBB    00      00  00      00
PPPPPPPP     AA      AA    SSSSSS  RRRRRRRR    EEEEEEEE      AA      AA  BBBBBBBB    00      00  00      00
PP           AAAAAAAAAA          SS  RR   RR    EE            AAAAAAAAAA  BB      BB  00      00  00      00
PP           AAAAAAAAAA          SS  RR   RR    EE            AAAAAAAAAA  BB      BB  00      00  00      00
PP           AA      AA          SS  RR    RR  EE            AA      AA  BB      BB  00      00  00      00
PP           AA      AA          SS  RR    RR  EE            AA      AA  BB      BB  00      00  00      00
PP           AA      AA  SSSSSSSS    RR      RR  EEEEEEEEEE  AA      AA  BBBBBBBB      000000      000000
PP           AA      AA  SSSSSSSS    RR      RR  EEEEEEEEEE  AA      AA  BBBBBBBB      000000      000000

LL             IIIIII      SSSSSSSS
LL             IIIIII      SSSSSSSS
LL               II      SS
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 MODULE PAS$READ_BOOLEAN ( %TITLE 'Read a boolean'
;     2          0002  0                      IDENT = '1-001'                    ! File: PASREAB00.B32 Edit: SBL100'
;     3          0003  0                      ) =
;     4          0004  1 BEGIN
;     5          0005  1 !
;     6          0006  1 !*****************************************************************************
;     7          0007  1 !*                                                                           *
;     8          0008  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;     9          0009  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;    10          0010  1 !*  ALL RIGHTS RESERVED.                                                      *
;    11          0011  1 !*                                                                           *
;    12          0012  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
;    13          0013  1 !*  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE           *
;    14          0014  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER      *
;    15          0015  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
;    16          0016  1 !*  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY        *
;    17          0017  1 !*  TRANSFERRED.                                                              *
;    18          0018  1 !*                                                                           *
;    19          0019  1 !*  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE     *
;    20          0020  1 !*  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT         *
;    21          0021  1 !*  CORPORATION.                                                              *
;    22          0022  1 !*                                                                           *
;    23          0023  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS     *
;    24          0024  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                   *
;    25          0025  1 !*                                                                           *
;    26          0026  1 !*                                                                           *
;    27          0027  1 !*****************************************************************************
;    28          0028  1 !
;    29          0029  1
;    30          0030  1 !++
;    31          0031  1 ! FACILITY:     Pascal Language Support
;    32          0032  1 !
;    33          0033  1 ! ABSTRACT:
;    34          0034  1 !
;    35          0035  1 !       This module contains procedures which read a boolean
;    36          0036  1 !       from a textfile or a string.
;    37          0037  1 !
;    38          0038  1 ! ENVIRONMENT:  User mode - AST reentrant
;    39          0039  1 !
;    40          0040  1 ! AUTHOR: Steven B. Lionel, CREATION DATE: 1-April-1981
;    41          0041  1 !
;    42          0042  1 ! MODIFIED BY:
;    43          0043  1 !
;    44          0044  1 ! 1-001 - Original.  SBL 1-April-1981
;    45          0045  1 !--
;    46          0046  1
```

```
;    48          0047  1 %SBTT_ 'Declarations'
;    49          0048  1 !
;    50          0049  1 ! PROLOGUE DEFINITIONS:
;    51          0050  1 !
;    52          0051  1
;    53          0052  1 REQUIRE 'RTLIN:PASPROLOG';                    ! Externals, linkages, PSECTs, structures
;    54          0116  1
;    55          0117  1 !
;    56          0118  1 ! TABLE OF CONTENTS:
;    57          0119  1 !
;    58          0120  1
;    59          0121  1 FORWARD ROUTINE
;    60          0122  1     PAS$READ_BOOLEAN,                        ! Read from textfile
;    61          0123  1     PAS$READV_BOOLEAN;                       ! Read from string
;    62          0124  1
;    63          0125  1 !
;    64          0126  1 ! MACROS:
;    65          0127  1 !
;    66          0128  1 !       NONE
;    67          0129  1 !
;    68          0130  1 ! EQUATED SYMBOLS:
;    69          0131  1 !
;    70          0132  1 !       NONE
;    71          0133  1 !
;    72          0134  1 ! FIELDS:
;    73          0135  1 !
;    74          0136  1 !       NONE
;    75          0137  1 !
;    76          0138  1 ! OWN STORAGE.
;    77          0139  1 !
;    78          0140  1
;    79          0141  1 !+
;    80          0142  1 ! Declare our own Pascal Enumerated Type Descriptor (PETD) for the BOOLEAN
;    81          0143  1 ! type.  This will be passed to PAS$READ_ENUMERATED.
;    82          0144  1 !-
;    83          0145  1
;    84          0146  1 GLOBAL
;    85          0147  1     PAS$$GR_BOOLEAN_PETD: VECTOR [4, LONG] PSECT (_PAS$CODE) INITIAL (
;    86          0148  1
;    87          0149  1         (UPLIT BYTE (%ASCIC'BOOLEAN ') - PAS$$GR_BOOLEAN_PETD),   ! Name of type
;    88          0150  1         2,                                                        ! Count of values
;    89          0151  1         (UPLIT BYTE (%ASCIC'FALSE') - PAS$$GR_BOOLEAN_PETD),      ! Value FALSE = 0
;    90          0152  1         (UPLIT BYTE (%ASCIC'TRUE') - PAS$$GR_BOOLEAN_PETD));      ! Value TRUE = 1
;    91          0153  1
```

```
;    93          0154  1 %SBTTL 'PAS$READ_BOOLEAN - Read a boolean from textfile'
;    94          0155  1 GLOBAL ROUTINE PAS$READ_BOOLEAN
;    95          0156  1         PFV: REF $PAS$PFV_FILE_VARIABLE,                ! File variable
;    96          0157  1         ERROR                                           ! Error unwind address
;    97          0158  1     ) =
;    98          0159  1
;    99          0160  1 !++
;   100          0161  1 ! FUNCTIONAL DESCRIPTION:
;   101          0162  1 !
;   102          0163  1 !       This function reads a boolean from the specified textfile
;   103          0164  1 !       and returns it as the function value.
;   104          0165  1 !
;   105          0166  1 ! CALLING SEQUENCE:
;   106          0167  1 !
;   107          0168  1 !       BOOLEAN.wv.v = PAS$READ_BOOLEAN (PFV.mr.r [, ERROR.ja.r])
;   108          0169  1 !
;   109          0170  1 ! FORMAL PARAMETERS:
;   110          0171  1 !
;   111          0172  1 !       PFV             - The Pascal File Variable (PFV) passed by reference.
;   112          0173  1 !                         The structure of the PFV is defined in PASPFV.REQ.
;   113          0174  1 !
;   114          0175  1 !       ERROR           - Optional.  If specified, the address to unwind to
;   115          0176  1 !                         in case of an error.
;   116          0177  1 !
;   117          0178  1 ! IMPLICIT INPUTS:
;   118          0179  1 !
;   119          0180  1 !       NONE
;   120          0181  1 !
;   121          0182  1 ! IMPLICIT OUTPUTS:
;   122          0183  1 !
;   123          0184  1 !       NONE
;   124          0185  1 !
;   125          0186  1 ! ROUTINE VALUE:
;   126          0187  1 !
;   127          0188  1 !       The value of the boolean read.
;   128          0189  1 !
;   129          0190  1 ! SIDE EFFECTS:
;   130          0191  1 !
;   131          0192  1 !       If the file is the standard file INPUT or OUTPUT, it is implicitly opened.
;   132          0193  1 !
;   133          0194  1 ! SIGNALLED ERRORS.
;   134          0195  1 !
;   135          0196  1 !
;   136          0197  1 !--
;   137          0198  1
;   138          0199  2     BEGIN
;   139          0200  2
;   140          0201  2     LOCAL
;   141          0202  2         PFV_ADDR: VOLATILE,                 ! Enable argument
;   142          0203  2         UNWIND_ACT: VOLATILE,               ! Enable argument
;   143          0204  2         ERROR_ADDR: VOLATILE;               ! Enable argument
;   144          0205  2
;   145          0206  2     BUILTIN
;   146          0207  2         ACTUALCOUNT;                        ! Count of arguments
;   147          0208  2
;   148          0209  2     ENABLE
;   149          0210  2         PAS$$IO_HANDLER (PFV_ADDR, UNWIND_ACT, ERROR_ADDR);     ! Enable error handler
```

```
;  150          0211  2
;  151          0212  2      !+
;  152          0213  2      ! Get ERROR parameter, if present.
;  153          0214  2      !-
;  154          0215  2
;  155          0216  2      IF ACTUALCOUNT () GEQU 2
;  156          0217  2      THEN
;  157          0218  2          ERROR_ADDR = .ERROR;            ! Set unwind address
;  158          0219  2
;  159          0220  2      !+
;  160          0221  2      ! Set PFV address.
;  161          0222  2      !-
;  162          0223  2
;  163          0224  2      PFV_ADDR = PFV [PFV$R_PFV];
;  164          0225  2
;  165          0226  2      !+
;  166          0227  2      ! Call READ_ENUMERATED to do the work.
;  167          0228  2      !-
;  168          0229  2
;  169          0230  3      RETURN (PAS$READ_ENUMERATED (
;  170          0231  3                  PFV [PFV$R_PFV],
;  171          0232  2                  PAS$$GR_BOOLEAN_PETD));
;  172          0233  2
;  173          0234  2
;  174          0235  1      END;                                        ! End of routine PAS$READ_BOOLEAN


                                                                         .TITLE  PAS$READ_BOOLEAN Read a boolean
                                                                         .IDENT  \1-001\

                                                                         .PSECT  _PAS$CODE,NOWRT,  SHR,  PIC,2

                                                 4E 41 45 4C 4F 4F 42 07 00000 P.AAA:    .ASCII  <7>\BOOLEAN\                    ;
                                                    45 53 4C 41 4C 53 05 00008 P.AAB:    .ASCII  <5>\FALSE\                      ;
                                                       45 55 52 54 04 0000E P.AAC:    .ASCII  <4>\TRUE\                       ;
                                                                     00013              .BLKB   1                               ;
                                                             00000000* 00014 PAS$$GR_BOOLEAN_PETD::
                                                                         .LONG   <P.AAA-PAS$$GR_BOOLEAN_PETD>                    ;
                                                             00000002  00018            .LONG   2                               ;
                                                   00000000* 00000000* 0001C            .LONG   <P.AAB-PAS$$GR_BOOLEAN_PETD>, <P.AAC--  ;
                                                                                                PAS$$GR_BOOLEAN_PETD>           ;

                                                                         .EXTRN  PAS$READ_BOOLEAN
                                                                         .EXTRN  PAS$READV_BOOLEAN
                                                                         .EXTRN  PAS$$IO_HANDLER
                                                                         .EXTRN  PAS$READ_ENUMERATED

                                                                0000 00000          .ENTRY  PAS$READ_BOOLEAN, Save nothing          ; 0155
                                                    5E                08  C2 00002         SUBL2   #8, SP                          ;
                                                                      7E  D4 00005         CLRL    ERROR_ADDR                      ; 0199
                                                              04      AE  7C 00007         CLRQ    UNWIND_ACT                      ;
                                                    6D      001D      CF  DE 0000A         MOVAL   2$, (FP)                        ;
                                                    02                6C  91 0000F         CMPB    (AP), #2                        ; 0216
                                                                      04  1F 00012         BLSSU   1$                              ;
                                                    6E          08    AC  D0 00014         MOVL    ERROR, ERROR_ADDR               ; 0218
                                                08  AE          04    AC  D0 00018 1$:     MOVL    PFV, PFV_ADDR                   ; 0224
                                                                D0    AF  9F 0001D         PUSHAB  PAS$$GR_BOOLEAN_PETD            ; 0231
```

```
                                 04   AC  DD 00020          PUSHL   PFV                                    ;
                      00000000G  00        02  FB 00023    CALLS   #2, PAS$READ_ENUMERATED                ;
                                           04  0002A       RET                                            ; 0235
                                         0000  0002B 2$:    .WORD   Save nothing                           ; 0199
                             50   08   AC  DO  0002D       MOVL    8(AP), R0                              ;
                             50   04   A0  DO  00031       MOVL    4(R0), R0                              ;
                                  F4   A0  9F  00035       PUSHAB  ERROR_ADDR                             ;
                                  F8   A0  9F  00038       PUSHAB  UNWIND_ACT                             ;
                                  FC   A0  9F  0003B       PUSHAB  PFV_ADDR                               ;
                                       03  DD  0003E       PUSHL   #3                                     ;
                                       5E  DD  00040       PUSHL   SP                                     ;
                             7E   04   AC  7D  00042       MOVQ    4(AP), -(SP)                           ;
                      00000000G  00        03  FB 00046    CALLS   #3, PAS$$IO_HANDLER                    ;
                                           04  0004D       RET                                            ;

; Routine Size:  78 bytes,    Routine Base:  _PAS$CODE + 0024

;  175          0236  1
;  176          0237  1 !<BLF/PAGE>
```

```
;  178          0238  1 %SBTTL 'PAS$READV_BOOLEAN - Read a boolean from a string'
;  179          0239  1 GLOBAL ROUTINE PAS$READV_BOOLEAN (                        ! Read an integer
;  180          0240  1         STRING: REF BLOCK [, BYTE],                      ! String descriptor
;  181          0241  1         ERROR                                            ! Error unwind address
;  182          0242  1     ) =
;  183          0243  1
;  184          0244  1 !++
;  185          0245  1 ! FUNCTIONAL DESCRIPTION:
;  186          0246  1 !
;  187          0247  1 !       This function reads a boolean from the specified string
;  188          0248  1 !       and returns it as the function value.
;  189          0249  1 !
;  190          0250  1 ! CALLING SEQUENCE:
;  191          0251  1 !
;  192          0252  1 !       Boolean.wv.v = PAS$READV_BOOLEAN (STRING.mt.ds [, ERROR.ja.r])
;  193          0253  1 !
;  194          0254  1 ! FORMAL PARAMETERS:
;  195          0255  1 !
;  196          0256  1 !       STRING          - The string to read from, passed as a class S
;  197          0257  1 !                         (assumed) descriptor.  The length and pointer
;  198          0258  1 !                         are updated to reflect the unread string.
;  199          0259  1 !
;  200          0260  1 !       ERROR           - Optional.  If specified, the address to unwind to
;  201          0261  1 !                         in case of an error.
;  202          0262  1 !
;  203          0263  1 ! IMPLICIT INPUTS:
;  204          0264  1 !
;  205          0265  1 !       NONE
;  206          0266  1 !
;  207          0267  1 ! IMPLICIT OUTPUTS:
;  208          0268  1 !
;  209          0269  1 !       NONE
;  210          0270  1 !
;  211          0271  1 ! ROUTINE VALUE:
;  212          0272  1 !
;  213          0273  1 !       The value of the boolean read.
;  214          0274  1 !
;  215          0275  1 ! SIDE EFFECTS:
;  216          0276  1 !
;  217          0277  1 !       NONE
;  218          0278  1 !
;  219          0279  1 ! SIGNALLED ERRORS:
;  220          0280  1 !
;  221          0281  1 !       NONE
;  222          0282  1 !
;  223          0283  1 !--
;  224          0284  1
;  225          0285  2     BEGIN
;  226          0286  2
;  227          0287  2     LOCAL
;  228          0288  2         PFV: $PAS$PFV_FILE_VARIABLE,     ! Pascal File Variable
;  229          0289  2         RESULT,                          ! Result value
;  230          0290  2         ARG_LIST: VECTOR [2, LONG],      ! Argument list
;  231          0291  2         PFV_ADDR: VOLATILE,              ! Enable argument
;  232          0292  2         UNWIND_ACT: VOLATILE,            ! Enable argument
;  233          0293  2         ERROR_ADDR: VOLATILE;            ! Enable argument
;  234          0294  2
```

```
;  235          0295  2     BUILTIN
;  236          0296  2         ACTUALCOUNT;                        ! Count of arguments
;  237          0297  2
;  238          0298  2     ENABLE
;  239          0299  2         PAS$$IO_HANDLER (PFV_ADDR, UNWIND_ACT, ERROR_ADDR);     ! Enable error handler
;  240          0300  2
;  241          0301  2     !+
;  242          0302  2     ! Get ERROR parameter, if present.
;  243          0303  2     !-
;  244          0304  2
;  245          0305  2     IF ACTUALCOUNT () GEQU 2
;  246          0306  2     THEN
;  247          0307  2         ERROR_ADDR = .ERROR;            ! Set unwind address
;  248          0308  2
;  249          0309  2     PFV_ADDR = PFV [PFV$R_PFV];         ! Set PFV address
;  250          0310  2
;  251          0311  2     !+
;  252          0312  2     ! Set up ARG_LIST.
;  253          0313  2     !-
;  254          0314  2
;  255          0315  2     ARG_LIST [0] = 1;                   ! One argument
;  256          0316  2     ARG_LIST [1] = PFV [PFV$R_PFV];     ! PFV address
;  257          0317  2
;  258          0318  2     !+
;  259          0319  2     ! Call PAS$$DO_READV to do the work, giving it the address of
;  260          0320  2     ! PAS$READ_BOOLEAN to call
;  261          0321  2     !-
;  262          0322  2
;  263          0323  2     PAS$$DO_READV (PFV [PFV$R_PFV], .STRING, ARG_LIST, PAS$READ_BOOLEAN;
;  264          0324  2         RESULT);
;  265          0325  2
;  266          0326  2     RETURN .RESULT;
;  267          0327  2
;  268          0328  1     END;                                ! End of routine PAS$READV_BOOLEAN


                                                .EXTRN  PAS$$DO_READV

                                 005C 00000     .ENTRY  PAS$READV_BOOLEAN, Save R2,R3,R4,R6     ; 0239
                      5E             20  C2 00002     SUBL2   #32, SP                         ;
                                     7E  D4 00005     CLRL    ERROR_ADDR                      ; 0285
                              04     AE  7C 00007     CLRQ    UNWIND_ACT                      ;
                      6D      002F   CF  DE 0000A     MOVAL   2$, (FP)                        ;
                      02             6C  91 0000F     CMPB    (AP), #2                        ; 0305
                                     04  1F 00012     BLSSU   1$                              ;
                      6E        08   AC  D0 00014     MOVL    ERROR, ERROR_ADDR               ; 0307
                08    AE        14   AE  9E 00018 1$: MOVAB   PFV, PFV_ADDR                   ; 0309
                0C    AE             01  D0 0001D     MOVL    #1, ARG_LIST                    ; 0315
                10    AE        14   AE  9E 00021     MOVAB   PFV, ARG_LIST+4                 ; 0316
                      54        89   AF  9E 00026     MOVAB   PAS$READ_BOOLEAN, R4            ; 0323
                      53        0C   AE  9E 0002A     MOVAB   ARG_LIST, R3                    ;
                      56        14   AE  9E 0002E     MOVAB   PFV, R6                         ;
                      52        04   AC  D0 00032     MOVL    STRING, R2                      ;
                          00000000G  00  16 00036     JSB     PAS$$DO_READV                   ;
                                     04  00003C     RET                                     ; 0328
                                 0000 0003D 2$:      .WORD   Save nothing                    ; 0285
```

```
                                 50       08    AC  D0 0003F          MOVL    8(AP), R0
                                 50       04    A0  D0 00043          MOVL    4(R0), R0
                                          DC    A0  9F 00047          PUSHAB  ERROR_ADDR
                                          E0    A0  9F 0004A          PUSHAB  UNWIND_ACT
                                          E4    A0  9F 0004D          PUSHAB  PFV_ADDR
                                                03  DD 00050          PUSHL   #3
                                                5E  DD 00052          PUSHL   SP
                                 7E       04    AC  7D 00054          MOVQ    4(AP), -(SP)
                       00000000G 00             03  FB 00058          CALLS   #3, PAS$$IO_HANDLER
                                                    04 0005F          RET

; Routine Size:  96 bytes,    Routine Base:  _PAS$CODE + 0072

;  269          0329  1
;  270          0330  1 !<BLF/PAGE>
```

```
;  272          0331  1 END                                        ! End of module PAS$READ_BOOLEAN
;  273          0332  1
;  274          0333  0 ELUDOM
```

```
;                                  PSECT SUMMARY

;       Name                    Bytes                           Attributes

;  _PAS$CODE                       210  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(2)
```

```
;                                Library Statistics

;                                          --------  Symbols  --------      Pages       Processing
;       File                               Total     Loaded   Percent       Mapped      Time

;  _$255$DUA28:[SYSLIB]STARLET.L32;1        9776          0         0          581        00:01.1
;  _$255$DUA28:[PASRTL.OBJ]PASLIB.L32;1      427         26         6           33        00:00.4
```

```
;                                  COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LIS$:PASREABOO/OBJ=OBJ$:PASREABOO MSRC$:PASREABOO/UPDATE=(ENH$:PASREABOO
;       )

;  Size:          174 code + 36 data bytes
;  Run Time:          00:05.5
;  Elapsed Time:      00:21.1
;  Lines/CPU Min:      3645
;  Lexemes/CPU-Min:    8605
;  Memory Used:  47 pages
;  Compilation Complete
```

PASREABOO
LIS

PASREAST1
LIS

PASREAREH
LIS

PASRESETK
LIS

PASREAREG
LIS

PASRAB
LIS

PASRESET2
LIS

PASREADLN
LIS

PASREAREF
LIS

PASREAVAR
LIS

PASREARED
LIS

PASREACHA
LIS

PASREADUT
LIS

PASREAENU
LIS

PASREAUNS
LIS

PASREAINT
LIS

PASREASTR
LIS