```
BBBBBBBBBBBB        AAAAAAAAA        SSSSSSSSSSSS    RRRRRRRRRRRR    TTTTTTTTTTTTTTT  LLL
BBBBBBBBBBBB        AAAAAAAAA        SSSSSSSSSSSS    RRRRRRRRRRRR    TTTTTTTTTTTTTTTTT LLL
BBBBBBBBBBBB        AAAAAAAAA        SSSSSSSSSSSS    RRRRRRRRRRRR    TTTTTTTTTTTTTTTT  LLL
BBB         BBB  AAA         AAA  SSS              RRR         RRR       TTT           LLL
BBB         BBB  AAA         AAA  SSS              RRR         RRR       TTT           LLL
BBB         BBB  AAA         AAA  SSS              RRR         RRR       TTT           LLL
BBB         BBB  AAA         AAA  SSS              RRR         RRR       TTT           LLL
BBB         BBB  AAA         AAA  SSS              RRR         RRR       TTT           LLL
BBBBBBBBBBBB     AAA         AAA     SSSSSSSSS     RRRRRRRRRRRR         TTT           LLL
BBBBBBBBBBBB     AAA         AAA     SSSSSSSSS     RRRRRRRRRRRR         TTT           LLL
BBBBBBBBBBBB     AAA         AAA     SSSSSSSSS     RRRRRRRRRRRR         TTT           LLL
BBB         BBB  AAAAAAAAAAAAAAAAA        SSS   RRR   RRR              TTT           LLL
BBB         BBB  AAAAAAAAAAAAAAAAA        SSS   RRR   RRR              TTT           LLL
BBB         BBB  AAAAAAAAAAAAAAAAA        SSS   RRR   RRR              TTT           LLL
BBB         BBB  AAA         AAA          SSS   RRR         RRR        TTT           LLL
BBB         BBB  AAA         AAA          SSS   RRR         RRR        TTT           LLL
BBB         BBB  AAA         AAA          SSS   RRR         RRR        TTT           LLL
BBBBBBBBBBBB     AAA         AAA  SSSSSSSSSSSS  RRR         RRR        TTT   LLLLLLLLLLLLLLL
BBBBBBBBBBBB     AAA         AAA  SSSSSSSSSSSS  RRR         RRR        TTT   LLLLLLLLLLLLLLL
BBBBBBBBBBBB     AAA         AAA  SSSSSSSSSSSS  RRR         RRR        TTT   LLLLLLLLLLLLLLL
```

BASINSTR.

LIS

```
    1   0001   0  MODULE BAS$INSTR (                              ! Find the position of a substring
    2   0002   0                 IDENT = '1-005'                  ! File: BASINSTR.B32
    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: String support library
   32   0032   1  !
   33   0033   1  ! ABSTRACT: This module takes 2 input strings of any class and dtype
   34   0034   1  !           and returns the position of the substring in the main string
   35   0035   1  !           starting at an input position
   36   0036   1  !
   37   0037   1  ! ENVIRONMENT: User mode, AST level or not or mixed
   38   0038   1  !
   39   0039   1  ! AUTHOR: R. Will, CREATION DATE: 10-Mar-79
   40   0040   1  !
   41   0041   1  ! MODIFIED BY:
   42   0042   1  !
   43   0043   1  ! R. Will, 10-Mar-79 : VERSION 01
   44   0044   1  ! 01    - original
   45   0045   1  ! 1-002 - String cleanup.  Call STR$POS.  RW  20-OCT-79
   46   0046   1  ! 1-003 - Correct a typo in the linkage specifications.
   47   0047   1  !           JBS 02-NOV-1979
   48   0048   1  ! 1-004 - Use JSB entry point of STR_POSITION.  RW  2-Nov-79
   49   0049   1  ! 1-005 - Use CALL entry point of STR_POSITION.  RW  15-NOV-79
   50   0050   1  !--
   51   0051   1
   52   0052   1  !<BLF/PAGE>
```

```
    54        0053   1  !
    55        0054   1  ! SWITCHES:
    56        0055   1  !
    57        0056   1
    58        0057   1  SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
    59        0058   1
    60        0059   1  ! LINKAGES: NONE
    61        0060   1  !
    62        0061   1  !
    63        0062   1
    64        0063   1  !
    65        0064   1  ! TABLE OF CONTENTS:
    66        0065   1  !
    67        0066   1
    68        0067   1  FORWARD ROUTINE
    69        0068   1      BAS$INSTR;                                  ! find substring position
    70        0069   1
    71        0070   1  ! INCLUDE FILES:
    72        0071   1  !
    73        0072   1
    74        0073   1  REQUIRE 'RTLIN:RTLPSECT';                       ! Declare PSECTS code
    75        0074   1
    76        0169   1
    77        0170   1  ! MACROS: NONE
    78        0171   1  !
    79        0172   1  !
    80        0173   1  ! EQUATED SYMBOLS: NONE
    81        0174   1  !
    82        0175   1  !
    83        0176   1  ! PSECT DECLARATIONS
    84        0177   1  !
    85        0178   1  !
    86        0179   1
    87        0180   1  DECLARE_PSECTS (BAS);
    88        0181   1
    89        0182   1  ! OWN STORAGE: NONE
    90        0183   1  !
    91        0184   1  !
    92        0185   1  ! EXTERNAL REFERENCES:
    93        0186   1  !
    94        0187   1  !
    95        0188   1
    96        0189   1  EXTERNAL ROUTINE
    97        0190   1      STR$POSITION;                               ! routine to do the actual search
    98        0191   1
```

```
100   0192  1  GLOBAL ROUTINE BAS$INSTR (                         ! find position of substring in string
101   0193  1          START_POS,                                 ! value of position to start search
102   0194  1          SRC_DESC,                                  ! descriptor of string to search
103   0195  1          SUB_DESC                                   ! descriptor of string to find
104   0196  1      ) : =                                          !
105   0197  1
106   0198  1  !++
107   0199  1  ! FUNCTIONAL DESCRIPTION:
108   0200  1  !
109   0201  1  !     This routine takes two source strings of any
110   0202  1  !     dtype and class and finds the position of the substring in the
111   0203  1  !     source string starting at the input starting position.  The routine
112   0204  1  !     returns the position of the substring in the source string. This
113   0205  1  !     routine merely calls BAS$POS which does the exact same thing with
114   0206  1  !     the parameters in a different order
115   0207  1  !
116   0208  1  ! FORMAL PARAMETERS:
117   0209  1  !
118   0210  1  !     START_POS.rl.v            value of position in source to begin search
119   0211  1  !     SRC_DESC.rt.dx           pointer to descriptor of string to be searched
120   0212  1  !     SUB_DESC.rt.dx           pointer to descriptor of string to find
121   0213  1  !
122   0214  1  ! IMPLICIT INPUTS:
123   0215  1  !
124   0216  1  !     NONE
125   0217  1  !
126   0218  1  ! IMPLICIT OUTPUTS:
127   0219  1  !
128   0220  1  !     NONE
129   0221  1  !
130   0222  1  ! ROUTINE VALUE:
131   0223  1  !
132   0224  1  !     FIND_POS.wlu.v            value of start of substring in source string
133   0225  1  !
134   0226  1  ! SIDE EFFECTS:
135   0227  1  !
136   0228  1  !     This routine calls STR$POSITION and therefore may signal any of
137   0229  1  !     its errors or have any of its side effects, including locking
138   0230  1  !     a string from being written for a short period.
139   0231  1  !
140   0232  1  !--
141   0233  1
142   0234  2     BEGIN
143   0235  2     RETURN STR$POSITION (.SRC_DESC, .SUB_DESC, START_POS);
144   0236  1     END;                                      !End of BAS$INSTR
```

```
                                          .TITLE   BAS$INSTR
                                          .IDENT   \1-005\

                                          .EXTRN   STR$POSITION

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

                      0000 00000          .ENTRY   BAS$INSTR, Save nothing
              04  AC  9F 00002            PUSHAB   START_POS
      7E      08  AC  7D 00005            MOVQ     SRC_DESC, -(SP)
```

```
            00000000G  00        03 FB 00009        CALLS   #3, STR$POSITION
                                 04 00010           RET
```
; 0236

; Routine Size: 17 bytes,   Routine Base: _BAS$CODE + 0000


```
; 145        0237  1
; 146        0238  1 END                                    !End of module
; 147        0239  1
; 148        0240  0 ELUDOM
```


;
;
;   PSECT SUMMARY
;
;
;   Name   Bytes   Attributes
;
; _BAS$CODE   17 NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

;   COMMAND QUALIFIERS

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

; Size: 17 code + 0 data bytes
; Run Time: 00:01.6
; Elapsed Time: 00:03.4
; Lines/CPU Min: 9056
; Lexemes/CPU-Min: 20981
; Memory Used: 17 pages
; Compilation Complete

BASINIGSC
LIS

BASINIT
LIS

BASINIDEF
LIS

BASINIDFS
LIS

BASINIGSB
LIS

BASINSTR
LIS

BASINIONE
LIS

BASLEFT
LIS

BASMARGIN
LIS

BASINIIOL
LIS

BASKILL
LIS

BASIOBEG
LIS

BASIOEND
LIS

BASMATADD
LIS

BASMAGTAP
LIS