```
c===========================================================
c     fdemo2:  Program which demonstrates basic usage
c     of character variables in Fortran 77.
c===========================================================
      program         fdemo2

      implicit        none
c-----------------------------------------------------------
c     See below for definition of integer function
c     'indlnb'.   Note that this and other useful routines
c     are available in the 'p329f' library.
c-----------------------------------------------------------
      integer         indlnb

c-----------------------------------------------------------
c     Define some character variables of various lengths
c-----------------------------------------------------------
      character*1     c1
      character*2     c2
      character*4     c4
      character*26    lcalph
      character       cc1*1,    cc2*2,   cc4*4
      character*60    buffer

c-----------------------------------------------------------
c     Assignment of constant strings to char. variables.
c     If length of character expression being assigned
c     is less than length of character variable, variable
c     is 'right-padded' with blanks.
c-----------------------------------------------------------
      c1     = 'a'
      c2     = 'bc'
      c4     = 'defg'
      lcalph = 'abcdefghijklmnopqrstuvwxyz'

      write(*,*) 'c1 = ', c1
      write(*,*) 'c2 = ', c2
      write(*,*) 'c4 = ', c4
      write(*,*) 'lcalph = ', lcalph
      call prompt('Through constant assignment')
```

```
c-------------------------------------------------------------
c     // is the string concatentation operator
c-------------------------------------------------------------
      write(*,*) 'c1 // c2 // c4 = ', c1 // c2 // c4
      call prompt('Through concatenation')

c-------------------------------------------------------------
c     The integer intrinsic (built-in) function 'len'
c     returns the length of its string argument
c-------------------------------------------------------------
      write(*,*) 'len(c1) = ', len(c1)
      write(*,*) 'len(buffer) = ', len(buffer)
      call prompt('Through string length')

c-------------------------------------------------------------
c     Substring extraction
c-------------------------------------------------------------
      write(*,*) 'lcalph(1:13) = ',  lcalph(1:13)
      write(*,*) 'lcalph(18:18) = ', lcalph(18:18)
      call prompt('Through substring extraction')

c-------------------------------------------------------------
c     Substring assignment
c-------------------------------------------------------------
      c4(4:4) = 'Z'
      write(*,*) 'c4 = ', c4
      call prompt('Through substring assignment')

c-------------------------------------------------------------
c     Use of 'indlnb'
c-------------------------------------------------------------
      buffer = 'somefilename'
      write(*,*) '<' // buffer // '>'
      write(*,*) '<' // buffer(1:indlnb(buffer)) // '>'
      buffer = 'Some multi-word message'
      write(*,*) '<' // buffer // '>'
      write(*,*) '<' // buffer(1:indlnb(buffer)) // '>'
      buffer = ' '
      write(*,*) 'indlnb(buffer) = ', indlnb(buffer)
      call prompt('Through indlnb usage')
```

```
call prompt('Through fdemo2')

stop
end
```

```
c-----------------------------------------------------------
c     Prints a message on stdout and then waits for input
c     from stdin.
c-----------------------------------------------------------
      subroutine prompt(pstring)

         implicit      none

         character*(*) pstring
         integer       rc
         character*1   resp

         write(*,*) pstring
         write(*,*) 'Enter anything & <CR> to continue'
         read(*,*,iostat=rc,end=900)  resp
         return

 900     continue
         stop
      end

c-----------------------------------------------------------
c     Returns index of last non-blank character in 's',
c     or 0 if the string is completely blank.
c-----------------------------------------------------------
      integer function indlnb(s)

         character*(*)    s
         integer          i

         do indlnb = len(s) , 1 , -1
            if( s(indlnb:indlnb) .ne. ' ' ) return
         end do
         indlnb = 0

         return

      end
```

```
##############################################################
#     Output from fdemo2
##############################################################

gauss 21> fdemo2
 c1 = a
 c2 = bc
 c4 = defg
 lcalph = abcdefghijklmnopqrstuvwxyz
 Through constant assignment
 Enter anything & <CR> to continue
a
 c1 // c2 // c4 = abcdefg
 Through concatenation
 Enter anything & <CR> to continue
a
 len(c1) =           1
 len(buffer) =          60
 Through string length
 Enter anything & <CR> to continue
a
 lcalph(1:13) = abcdefghijklm
 lcalph(18:18) = r
 Through substring extraction
 Enter anything & <CR> to continue
a
 c4 = defZ
 Through substring assignment
 Enter anything & <CR> to continue
a
 <somefilename                                               >
 <somefilename>
 <Some multi-word message                                    >
 <Some multi-word message>
 indlnb(buffer) =           0
 Through indlnb usage
 Enter anything & <CR> to continue
a
 len('') =           0
 c1 // '' // c2 = abc
```

```
 Through null string
 Enter anything & <CR> to continue
a
 Through fdemo2
 Enter anything & <CR> to continue
a
```