NAME
> ENGTRN − Find the translations corresponding to a vector of transliterations.

SYNOPSIS
> **CALL ENGTRN(TLS,N,NUNIT, ENGS,RC)**

> TLS(N)   is the CHARACTER*18 vector of transliterations to translate
> N   is the INTEGER*4 number of transliterations in TLS
> NUNIT   is the INTEGER*4 unit number to use for the dictionary file
> ENGS(N)   is the CHARACTER*80 vector of translations
> RC   is the INTEGER*4 return code; see below

DESCRIPTION
> First the routine copies the input list of transliterations to an automatic local vector and alphabetizes the copy. Then it reads the dictionary file, finding the translation that corresponds to each transliteration in the sorted list (because the list is sorted this can be done by reading the file only once). As each translation is found it is stored in the original order of the transliterations.

DIAGNOSTICS
> If all of the transliterations are found in the dictionary then RC=0 on return; otherwise the first transliteration that is not found is TLS(RC).

UNITS and FILES
> NUNIT ${HOME}/Utility/hebrew.hsh
> > but only while this routine is active

LINKAGE
> gfortran   source.f   −L${HOME}/lib   −lmisc

AUTHOR
> Michael Kupferschmid

EXAMPLE

```
 CHARACTER*18 TLS(3)/'zavach            ',
;                     'achat             ',
;                     'banav             '/
 CHARACTER*80 ENGS(3)
 INTEGER*4 RC
 CALL ENGTRN(TLS,3,1, ENGS,RC)
 DO 1 I=1,3
      WRITE(6,901) TLS(I),ENGS(I)
901      FORMAT(A18,1X,A80)
  1 CONTINUE
 STOP
 END
```

This example produced the following output:

```
unix[1] a.out
zavach            he slaughtered as a sacrifice
achat             one/one of (f)
banav             his sons
unix[2]
```