Identification
The EPL run-time routine, strcmp_
strcmp_$1eb_
strcmp_$1ec_
strcmp_$1tb_
strcmp_$1tc_
strcmp_$eqb_
strcmp_$eqc_
strcmp_$neb_
strcmp_$nec_
strcmp_$gtb_
strcmp_$gtc_
strcmp_$geb_
strcmp_$qec_
Ruth A. Weiss

Purpose
Strcmp_ implements the PL/I comparison operations for strings.

Usage
The calls are listed below with statements listing their effect. B1 and b2 are bit strings and c1 and c2 are character strings. Strcmp_ accepts either varying or non-varying strings as arguments. The result (answer) of a comparison is a bit string of length one; the value is '1' b if the relationship is true or '0' b if it is false.

call strcmp_$1eb_(b1,b2,answer)
    answer=(b1<b2)
call strcmp_$1ec_(c1,c2,answer)
    answer=(c1<c2)
call strcmp_$1tb_(b1,b2,answer)
    answer=(b1<b2)
call strcmp_$1tc_(c1,c2,answer)
    answer=(c1<c2)
call strcmp_$eqb_(b1,b2,answer)
    answer=(b1=b2)
call strcmp$_eqc_ (c1, c2, answer)
    answer = (c1 = c2)
call strcmp$_neb_ (b1, b2, answer)
    answer = (b1 ≠ b2)
call strcmp$_nec_ (c1, c2, answer)
    answer = (c1 ≠ c2)
call strcmp$_gtb_ (b1, b2, answer)
    answer = (b1 > b2)
call strcmp$_gtc_ (c1, c2, answer)
    answer = (c1 > c2)
call strcmp$_geb_ (b1, b2, answer)
    answer = (b1 ≥ b2)
call strcmp$_gec_ (c1, c2, answer)
    answer = (c1 ≥ c2)

Errors
If any argument is not a string, will stop on oct 0.