

Mar 15, 23 11:54

toupr.s

Page 1/1

```

/**
 * toupr.s
 * PARAMETERS: X0 (INPUT STRING OF SIZE N)
 * OUTPUT      : X0 (STRING, CONVERTED)
 * The purpose of this function is to convert all lowercase letters
 * to uppercase.
 * ALL REGISTERS PRESERVED EXCEPT X0
 */

.text
.global toupr
    toupr:
        stp x19,x20, [sp, -16]! // pushing x19 and x20 to stack pointer
        mov x19, #-1           // setting increment
        b    toUprLoop         // branch to 'toUprLoop'
    toUprConvert:
        sub w20, w20, #32       // converts the char from lowercase to uppercase
        strb w20, [x0, x19]     // storing byte in w0 string
    toUprLoop:
        add x19, x19, #1        // x19++
        ldrb w20, [x0, x19]     // load byte from str and post increment
        cmp w20, #0             // compare loaded byte and null
        b.eq toUprEnd           // if null, goto 'toUprEnd'
        cmp w20, #'a'           // if char is less than 'a'...
        b.lt toUprLoop          // then return to toUprLoop
        cmp w20, #'z'           // else, if w20 is less than 'z'
        b.le toUprConvert       // you're lowercase, so go to the convert label
        b    toUprLoop          // you're one of the few weird chars, go back to
upr :)
    toUprEnd:
        ldp x19, x20, [sp], 16 // popping x19 and x20 from stack
        ret lr                 // return to caller

```