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

.text
.global tolwr
    tolwr:
        stp x19,x20, [sp, -16]! // pushing x19 and x20 to stack pointer
        mov x19, #-1            // setting increment
        b   toLwrLoop          // branch to 'toLwrLoop'
    toLwrConvert:
        add w20, w20, #32       // converts the char from uppercase to lowercase
        strb w20, [x0, x19]     // storing byte in w0 string
    toLwrLoop:
        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 toLwrEnd           // if null, goto 'toLwrEnd'
        cmp  w20, #'A'          // if char is less than 'Z'...
        b.lt toLwrLoop          // then return to toLwrLoop
        cmp  w20, #'Z'          // else, if w20 is less than 'Z'
        b.le toLwrConvert       // you're lowercase, so go to the convert label
        b    toLwrLoop          // go back to lwr :)
    toLwrEnd:
        ldp x19, x20, [sp], 16  // popping x19 and x20 from stack
        ret lr                  // return to caller
```