Example #18 -- ARM Nested Control Constructs

The program reads one line of text from standard input, then displays the number of characters and the number of blanks found.

.global main
.text
main:   push    {lr}
    ldr     r0, =fmt1   @ First arg -- address of format string
    bl      printf      @ Prompt the user to enter a line
    mov     r4, #0      @ Initialize character count
    mov     r5, #0      @ Initialize blank count
    loop:   bl      getchar     @ Read one character (returned in r0)
            add     r4, r4, #1  @ Increment character count
    if:     cmp     r0, #0x20   @ Compare return value to blank
            bne     endif       @ If blank not found, skip increment
    then:   add     r5, r5, #1  @ Increment blank count
    endif:  cmp     r0, #0x0a   @ Compare return value to newline
            beq     end         @ If newline found, exit loop
            b       loop        @ Branch to top of loop
    end:    ldr     r0, =fmt2   @ First arg  -- address of format string
            mov     r1, r4      @ Second arg -- character count
            mov     r2, r5      @ Third arg  -- blank count
            bl      printf      @ Display the two counts
    pop     {lr}
    bx      lr

fmt1:   .asciz  "\nEnter a line of text: "
fmt2:   .asciz  "\nThe line has %d characters, including %d blanks\n"

<2 lemon:~/Examples > gcc example18.s
<3 lemon:~/Examples > a.out
Enter a line of text: A short line!
The line has 14 characters, including 2 blanks
<4 lemon:~/Examples > a.out
Enter a line of text: A line with more characters.
The line has 29 characters, including 4 blanks