

Apr 25, 23 22:55	main.s	Page 1/3
<pre> .data prompt_str_one: .asciz "Enter String #1 - " prompt_str_two: .asciz "Enter String #2 - " disp_str_one: .asciz "Displaying string #1 - " disp_str_two: .asciz "Displaying string #2 - " convert_str: .asciz "Converting to Upper Case ..." nl: .asciz "\n" .text .global main main: // Save space on the stack for storing original strings sub sp, sp, 32 // First input ldr x0, =prompt_str_one bl putstr bl inprdr str x0, [sp, #16] // Second input ldr x0, =prompt_str_two bl putstr bl inprdr str x0, [sp] // Print original strings ldr x0, =disp_str_one bl putstr ldr x0, [sp, #16] bl putstr ldr x0, =disp_str_two bl putstr ldr x0, [sp] bl putstr ldr x0, =nl bl putstr //Print converting ldr x0, =convert_str bl putstr // Print newline ldr x0, =nl bl putstr // Convert first input to uppercase and print ldr x0, [sp, #16] bl str_toupper ldr x0, =disp_str_one bl putstr ldr x0, [sp, #16] bl putstr // Free the first input ldr x0, [sp, #16] bl free // Convert second input to uppercase and print ldr x0, [sp] bl str_toupper ldr x0, =disp_str_two </pre>		

Apr 25, 23 22:55	main.s	Page 2/3
<pre> bl putstr ldr x0, [sp] bl putstr // Clean up the second input ldr x0, [sp] bl free // Print newline ldr x0, =nl bl putstr // Restore the stack pointer and setup the parameters to exit the program add sp, sp, 32 mov x0, #0 // Sets return code to 0 mov x8, #93 // Service command code 93 terminates svc 0 // Call Linux to terminate the program /** * str_toupper - Converts a null-terminated string to upper case * * @param x0: Pointer to the null-terminated string * * Registers used: x0, w8 */ .text .global str_toupper str_toupper: // Loop through each character in the string str_toupper_loop: ldrb w8, [x0] // Load a byte from the string cmp w8, #'a' - 1 // Compare if character is greater than 'a' bhs not_lower // If not greater, skip conversion (not a lowercase letter) cmp w8, #'z' // Compare if character is less than 'z' bhi not_lower // If not less, skip conversion (not a lowercase letter) // Convert to uppercase sub w8, w8, #('a' - 'A') strb w8, [x0] // Store the uppercase character back into the string not_lower: cbz w8, exit // Exit the loop if the null-terminator is reached add x0, x0, #1 // Move to the next character b str_toupper_loop exit: ret /** * putstr - Prints a null-terminated string using write syscall * * @param x0: Pointer to the null-terminated string * * Registers used: x0, x2, x30, lr </pre>		

Apr 25, 23 22:55

main.s

Page 3/3

```
* Registers saved: lr
*/

.text
.global putstr
putstr:
    // Save the link register
    str lr, [sp, #-16]!

    // Prepare parameters for the 'write' syscall
    mov x2, #0
    // Get the string length
putstr_loop:
    ldrb w1, [x0, x2]
    cbz w1, out
    add x2, x2, #1
    b putstr_loop
out:
    mov x8, #64
    mov x1, x0
    mov w0, #1
    svc 0

    // Restore the link register
    ldr lr, [sp], #16

    ret
```