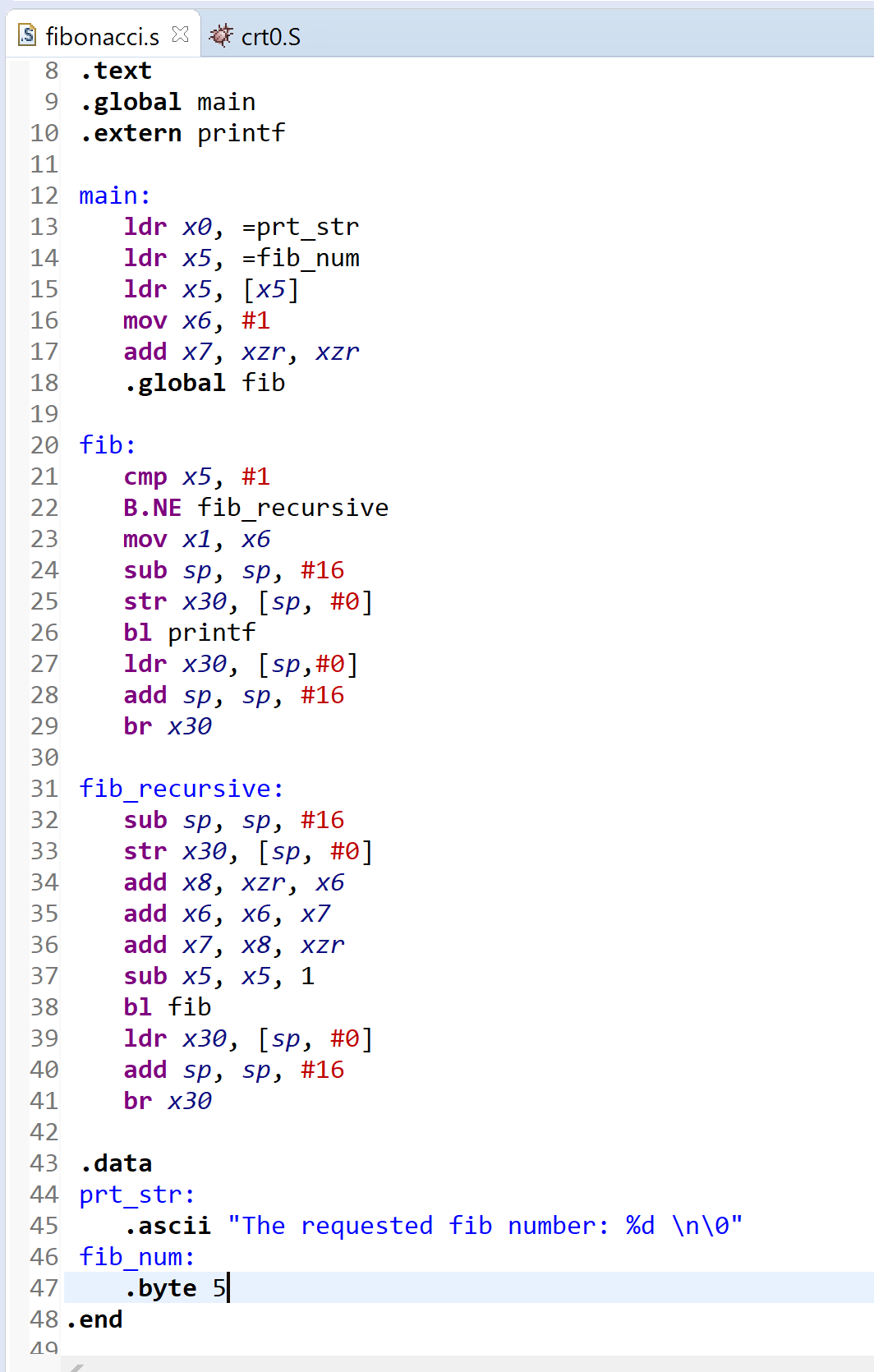
Siddharth Iyer

ARM Lab 1 – Fibonacci

Below is my full program for the Fibonacci function



.data contains the output string template: “The requested fib number: %d \n\0”

Also contains fibn, the desired output index (in this case, it is 5)

.global main – sends the start of the code to **main**

main:

* Loads the output string .ascii into x0
* the desired Fibonacci index is loaded into x5
* x6 = 1 (this is current)
* x7 = 0 (this is previous)
* the **fib** function is called

fib:

* if x5 == 1, current is moved to x1 as output and the stacks are closed / printf is also called for x0 and x1
* else: **fib\_recursive** is called

fib\_recursive:

* only need to store link register so only 1 extra position need on stack pointer (sub sp, sp, #16)
* add x8, xzr, x6 (temp = current) // none of this
* add x6, x6, x7 (current = current + previous) // needs to be
* add x7, x8, xzr (previous = old current) // stored on stack
* sub x5, x5, 1 (n = n - 1)
* bl fib (branch to fib to recheck n == 1 – described above in fib section)
* after return, destroy the stack
* printf does not work for me but I understand several people are also having the same issue
* it does not seem to be an issue during compilation

Screenshots of changing link registers:

