



Programming Lab 3B

[Click to download Lab3B-Main.c](#)

Functions & Parameters

Topics: Passing parameters, function return values, nested functions, preserving and restoring registers across function calls, calling C functions from assembly.

Prerequisite Reading: Chapters 1-3

Revised: January 1, 2022

Assignment: The main program contains each of the five functions shown below. The program may be compiled and executed without writing any assembly. However, your task is to create assembly language replacements for each of these C functions. The original C functions are defined as “weak”, so that the linker will automatically replace them in the executable image by those you create in assembly; you do not need to remove the C versions. Functions Factorial and gcd are provided in the main program; do not recreate them in assembly – just call them from within your assembly language code for functions FactSum32 and XPlusGCD.

<pre>int32_t Return32Bits(void) { return (int32_t) +10 ; }</pre>	<pre>uint8_t Add8Bits(uint8_t x, uint8_t y) { return (uint8_t) (x + y) ; }</pre>
<pre>int64_t Return64Bits(void) { return (int64_t) -10 ; }</pre>	<pre>uint32_t FactSum32(uint32_t x, uint32_t y) { return (uint32_t) Factorial(x + y) ; }</pre>
<pre>uint32_t XPlusGCD(uint32_t x, uint32_t y, uint32_t z) { return (uint32_t) (x + gcd(y, z)) ; }</pre>	

Code and test your functions one at a time using the main program downloaded using the link above. If your code works correctly, the display should look like the image shown. Press the blue pushbutton to cycle through all the test cases to verify that everything is correct. Errors are displayed in red.

IMPORTANT – The .thumb_func directive:

The ".thumb_func" assembler directive specifies that the next label is the entry point of a function that contains instructions from the Thumb subset of the ARM processor and causes the binary representation of instructions that branch to that label to be generated somewhat differently. Thus in a source code file that contains more than one function, it is imperative that you place a .thumb_func directive immediately before the entry point label of every function.

Note: The ARM Procedure Call Standard requires the stack pointer be double-word aligned when functions are called. Failure to observe this requirement can cause a program to behave incorrectly or crash.

