

Goal: Learning to write functions with loops and if statements in assembly language.

Background: This lab consists of two programs. Each program calls and tests a function. One program tests a function that computes the integer square root (isqrt) and the other tests a function that performs integer exponentiation (iexp). C versions of each function are provided; your assignment is to translate these functions into ARM assembly and run the program to verify that your implementation works.

PART I: Preparation

1. Find the folder called "Lab Assignments" on your desktop and open it.
2. Find a double click on the file called "COEN 20.eww" to open the IAR Embedded Workbench program.
3. If step 2 did not open IAR Embedded Workbench, find the program on the Start Menu and open it. Once open, click on "File > Open > Workspace", navigate to the "COEN 20.eww" file inside the "Lab Assignments" folder and open it.

PART II: Square root (using the code provided in file isqrt.c)

1. In the Workspace panel on the left side of the screen, right-click on "Lab08IntegerFunctionsWithLoops" and select "Set As Active".
2. Click on the "+" sign next to "Lab08IntegerFunctionsWithLoops".
3. Click on the "+" sign next to "Source".
4. Set each of the following source code files as follows:

a. isqrt.c	Include (do NOT check)
b. isqrt.s	exclude (CHECK)
c. iexp.c	exclude (CHECK)
d. iexp.s	exclude (CHECK)
e. main4isqrt.c	Include (do NOT check)
f. main4iexp.c	exclude (CHECK)
5. To compile the program, right-click on "Lab08IntegerFunctionsWithLoops" and select "Rebuild All". The build should complete with no errors or warnings.
6. Connect the LM3S811 board to a USB port on your computer. This provides both power (as indicated by the power LED) and a download connection to the device. From the drop-down menus, select "Project > Download > Download active application" to start the download.
7. To run the program, press the reset button (the one closest to the thumbwheel) on the LM3S811 evaluation board. Press the other button to sequence through all the test cases. Verify that it behaves as expected.

PART III: Square root (implementing your ARM code in file isqrt.s)

1. Change each of the following source code files as follows:
 - a. isqrt.c exclude (CHECK)
 - b. isqrt.s Include (do NOT check)
2. Edit isqrt.s to implement the algorithm described in isqrt.c.
3. Recompile the program, download it to the board, and verify that it works properly.
4. Demonstrate your working program and submit your source code of isqrt.s to the teaching assistant according to his instructions.

PART IV: Exponentiation (using the code provided in file iexp.c)

1. Change each of the following source code files as follows:
 - a. isqrt.s exclude (CHECK)
 - b. iexp.c Include (do NOT check)
 - c. main4isqrt.c exclude (CHECK)
 - d. main4iexp.c Include (do NOT check)
2. Recompile the program, download it to the board, and verify that it operates as expected.

PART V: Exponentiation (implementing your ARM code in file iexp.s)

1. Change each of the following source code files as follows:
 - a. iexp.c exclude (CHECK)
 - b. iexp.s Include (do NOT check)
2. Edit iexp.s to implement the algorithm described in iexp.c.
3. Recompile the program, download it to the board, and verify that it works properly.
4. Demonstrate your working program and submit your source code of iexp.s to the teaching assistant according to his instructions.