Computing Machinery I
Assignment 1

Basic Assembly Language Programming

Create an ARMv8 A64 assembly language program that finds the maximum of $y = 5x^3 - 12x^2 - 20x + 10$ in the range $-4 \leq x \leq 3$, by stepping through the range one by one in a loop and testing. Use only long integers for x, and do not factor the expression. Use the printf() function to display to the screen the values of x, y and the current *maximum* on each iteration of your loop.

You are to create 2 versions of your program:

1. Write the program without macros (i.e. don't use *m4*), and use only the *mul, add,* and *mov* instructions to do your calculations. Use a pre-test loop, where the test is at the top of the loop.
2. Optimize the above program by putting the loop test at the bottom of the loop (make sure it is still a pre-test loop), and by making use of the *madd* instruction. Also, add macros to the above program to make it more readable (use *m4*). In particular, provide macros for heavily used registers.

Running Your Program

To verify that your assembly language program works, run both versions under *gdb*, capturing output from each session using the *script* UNIX command. For version 1, single step through the program (use ni) for at least one iteration of your loop, displaying the instruction being executed (use display/i $pc). Also print out the contents of particular registers (use p) at key points in your program to show that it is working as expected. For version 2, set a breakpoint just after the place where the final result is calculated, and then print out the *maximum*. Do not single step through this version.

Other Requirements

Make sure your code is properly formatted into columns, is readable and fully documented, and includes identifying information at the top of each file. You must comment each line of assembly code. Your code should also be well designed: make sure it is well organized, clear, and concise.

New Skills Needed for this Assignment:

- Ability to work with basic arithmetic, loops, and if-else constructs in assembly
- Ability to print to standard output using the printf() function
- Ability to optimize assembly code by rearranging loops and using alternate instructions
- Ability to use macros in assembly code
- Ability to assemble programs using *gcc* and use *m4* to process macros
- Ability to use *gdb* to debug and display assembly language programs

Submit the following:

1. Your assembly source code files for both programs and 2 scripts via electronic submission. Use the *Assignment 1* Dropbox Folder in D2L to submit electronically. Your TA will assemble and run your programs to test them. Name your programs *assign1a.s* and *assign1b.asm*, and your scripts *script1.txt* and *script2.txt*.

Computing Machinery I
Assignment 1 Grading

Student:_____

Functionality (Version 1)

Equation calculation	4	_____
Test for maximum	4	_____
Display to screen using printf()	2	_____
Loop	4	_____
Optimization (Version 2)	3	_____
Use of Macros (Version 2)	3	_____
2 Scripts showing use of *gdb*	4	_____
Complete documentation and commenting	4	_____
Design quality	2	_____
Total	**30**	_____ _____%