CSE 2312 – Homework #2

Assigned: Thursday, June 15, 2017
Due: Thursday, June 22, 2017 at the end of class

Assignment:

1. Write an ARMv7 assembly program to perform the following operations in the order given. Numbers given will be base-10 unless specified otherwise.

   (a) store 13 in a register

   (b) store 110_2 in a register

   (c) subtract the number in 1b from 1a

   (d) multiply the result from 1c by the constant 14_{16}

   (e) divide the result from 1d by 4 by shifting the bits right the appropriate number of times

   (f) the final result should be in register R0

   The source code for this program will be named `hw02a.s`, all in lowercase.

2. Write an ARMv7 assembly program that begins summing the integers 20, 19, 18, . . ., stopping when the sum exceeds 105.

   (a) Your loop should begin at 20 and continue counting down in steps of 1 until the sum exceeds 105. Therefore, your loop will not know in advance how many iterations it will run.

   (b) The last number added to the sum, the one that caused the sum to exceed 105, should be in register R0.

   The source code for this program will be named `hw02b.s`, all in lowercase.

3. Write an ARMv7 assembly program that uses a loop to generate the integers from 1 to 15, inclusive, summing the odd integers in that range.

   (a) Your loop should generate all of the integers in the range of 1 to 15, not just the odd integers. This means you will need the logic that can determine which of the numbers are odd.

   (b) The final result should be in register R0.

   The source code for this program will be named `hw02c.s`, all in lowercase.

Note the following about each program:

1. You must comment your code. This means having enough comments that I can easily follow your logic.

2. Your program should conform to the assemble/link approach that we did in class and should have a `_start` section. That is, I should be able to do the following, assuming your program is called `hw02.s`:

```
as  -o hw02.o  hw02.s
ld  -o hw02  hw02.o
```

3. Note, if you submit code that I think was produced by a compiler, then you will not receive credit.

4. As a comment in your program, include your name.

5. To submit, create a directory with a name that matches your net ID (for example, `abc1234`) in lowercase.

6. Place your programs (all of them) in the directory you created, then tar and compress the directory. If the directory is called `abc1234`, then you would type

```
tar  cvzf  abc1234.tgz  abc1234
```

7. Upload the compressed and tar'd file to Blackboard.