Course: CDA-4205L Professor: Yan Zhang PhD

Assignment: Lab 7 Due: 12/02/21 at 11:59 pm

Description and Instructions

For this lab you will be looking at the RISC-V instruction set covered in chapters 2-4 of your textbook.

You will need to complete the following questions and show all work when necessary. You will also have a coding portions. For the coding portions you are required to create a lab report outlining any equation and calculations you performed in order to code your solutions. You will also need to answer some questions regarding your solution and provide some screenshots in your report. Feel free to use this document as a foundation for your report or create your own. All solutions need to be typed, no hand written solutions will be accepted.

To submit you will need to upload a single PDF file lab report which will include the answers to the questions below as well as sections outlining your solutions for the coding portion. You will also need to provide a section that includes the code you wrote. Make sure that the code is properly formatted and runs properly in the RISC-V simulation we covered in class.

All submissions should be done through Canvas by the due date or will be subjected to the penalties outlined in the syllabus. We encourage collaboration, however, you must submit your own original work must be submitted and cheating will not be tolerated. Your solutions will need to follow strict adherence to the RISC-V coding style. This means that you your solutions should be case sensitive, if commenting, make sure you use // to represent the commented section. A new line will be associated with a new line of code and use of indentation is needed to separate label, instructions, and registers (both destination and source).

RISC-V Code Problem

(10pts) 1**.** You will need to write a program that will generate the first *n* integers of the FibonacciSequence. Store the integers in an array starting at memory address **1028**.

Once you generate the first *n* integers you will need to sum all of the **even integers** and store the value at **memory address 1024**.

For example, if we consider the first 15 integers of the Fibonacci Sequence, say F_{15}.

F_{15} = {1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610}

We can see that the only even numbers are 2, 8, 34, 144, 610 and if we sum these together we will get 798.

```
addi x27, x0,1024
add x28,x0,x0
sw x28,0(x27)
addi x27,x27,4
addi x29, x0,1
sw x29,0(x27)
addi x5, x0, 15 # n element to stored- can be changed
addi s1, x5,0
add a0,x0,x0 #a0will store the result of the whole sequence
beq x5,x0, exit
addi a0,x29,0 #will store the result of the whole sequence
addi x30, x0,2
blt x5,x30,exit
add x31,x0,x0
addi x5,x5,-1
for:
beq x5,x0,sum
add x31,x28,x29
sw x31,4(x27)
addi x27,x27,4
add a0, x31,x0# register with result as before
add x28,x29,x0
add x29,x31,x0
addi x5,x5,-1
beq x0,x0,for
sum:
```

```
addi x27, x0,1024
addi s2, s2,4
mult s1, s1,s2
add s2, s1,x27
addi s6, s6, 1024
addi s7, x0,0

loop:

bge x27,s2,exit
lw s3, 0(s6)
addi x27, x27,12
lw s8, 0(x27)
add s9,s8,s3
sw s9,0(s6)



beq x0,x0,loop




exit:

nop
```