Homework 5 - Due in Discussion Wed Nov. 2nd

Instructions: You should do this homework in your group assigned to you in your 252 section. You should hand in ONE copy of the homework that lists the common section number and names and UW ID numbers of all students. You should **staple** multiple pages together.

Warning: Most homeworks will use questions from your textbook, Patt and Patel's *Introduction to Computing Systems*, which we abbreviate (*ItCS*) .

First contact for questions is TA Dustin Kreft (dkreft@wisc.edu)

Problem 1(4 points)

Explain the relationship, and their operation, between the PC and MAR registers as well as the IR and MDR registers during the FETCH cycle. What are their respective sizes in a system with 32 address bits with 64 bit addressability?

Problem 2(6 points)

What mathematical relationship do the registers R1 and R2 have in relationship to each other when looking at the result in R3 given the following sequence? Assume all register values are positive. (Please show your steps for solving this problem to receive full credit)

1) Convert the binary pattern to its Assembly mnemonics
2) Update/show the contents of the registry file and CC register for each step
3) List the operation being performed

Address	Instruction
x4400	0101011011100000
x4401	0001011011000001
x4402	0001010010111111
x4403	0000010111111101

Problem 3(5 points)

The program shown below is loaded into memory starting at address 0x30FF. After the program is run what is the value in register R2? Convert the binary into Assembly.

0x30FF	1110 0010 0000 0001
0x3100	0110 0100 0100 0010
0x3101	1111 0000 0010 0101
0x3102	0001 0100 0100 0001
0x3103	0001 0100 1000 0010

Problem 4(5 points)

State the contents of the registers R1, R2, R3, and R4 after the program shown below halts. The program starts at 0x3100:

Address	Data
0011 0001 0000 0000	1110 001 000100000
0011 0001 0000 0001	0010 010 000100000
0011 0001 0000 0010	1010 011 000100000
0011 0001 0000 0011	0110 100 010 000001
0011 0001 0000 0100	1111 0000 0010 0101
…	…
0011 0001 0010 0010	0100 0101 0110 0110
0011 0001 0010 0011	0100 0101 0110 0111
…	…
0100 0101 0110 0111	1010 1011 1100 1101
0100 0101 0110 1000	1111 1110 1101 0011

Problem 5(4 points)

Using only the instructions provided for LC-3 create the following assembly operations using only ONE LC-3 instruction:

a) MOV: This instruction moves the contents of one register to another without changing the source register.

b) CLR: This instruction clears the contents of a register setting its value to zero.

c) LSL: This instruction shifts each of the bits in a register to the left by one position.

d) NOP: This instruction does nothing and also does not change the contents of any register.

Problem 6(6 points)

The purpose of this problem is to get you setup with the PennSim LC-3 simulator, which will be important for subsequent homeworks.

To get started with PennSim, look at the information on the Computing page. Specifically, go through the PennSim Guide.

You should be able to run PennSim on any computer (with Java 1.5 or higher installed). They do not require any installation; they are self-contained executable files. You should be able to run them from any directory and even from a USB thumb drive

After going through the PennSim guide, please do the following:

1. Convert the capital letter initials of each team member's name into a 4-digit hexadecimal ASCII code. For example, if your name is Ryan Johnson, your initials are RJ. The hex ASCII code for R is 52 and for J is 4A. Therefore, the 4-digit hex code for Ryan Johnson's initials would be x524A.
2. Open PennSim and change the PC register to x3000. See the first paragraph in section 4 of the PennSim guide for how to set a register's value. (If you are feeling ambitious, write a script as discussed in section 5 of the guide to do this and the next step. This may be valuable in a later homework.) *Note:*You do not need to load the lc3os.obj object as discussed in the guide.
3. Beginning at memory address x3000, load each team member's initials hex code into memory. The method for setting the value of a memory location is the exact same as for setting a register's value.
4. Take a screen shot of PennSim showing your team member's initials in memory starting at address x3000 and turn it in for your answer. See the computing page if you need help taking a screenshot.