

Homework #2 for ECE 496

Instruction Sets (Chapter 2)

Hardcopy due in class on February 5th
Code due in Sakai at 10:00am on February 5th

All homework must be done in a group of 2 students. Each group should turn in one hardcopy in class. If your handwriting is unreadable, please type your homework.

1) [5] What MIPS instruction is this? 1010 1110 0000 1011 0000 0000 0000 0100

2) [15] Write two assembly programs (on paper) to compute $X = A * ((B + C) + (A + D + E) * (B + C))$. The first one should be based on a stack-based ISA and the result should be left on the top of the stack. The second one should be based on a register-based ISA and the result should be left in a memory location denominated by X. Assume that you have the following instructions:

Stack-based ISA

push A → puts A on top of stack
pop A → removes A from top of stack
add → consumes top two items of stack and puts sum on top of stack
mult → consumes top two items of stack and puts product on top

Register-based ISA

//assume you have four registers: \$r1, \$r2, \$r3, \$r4
lw \$rd, A → loads A in register \$rd
sw \$rs, X → stores \$rs in memory location X
add \$rd, \$rs, \$rt → $\$rd = \$rs + \$rt$
mult \$rd, \$rs, \$rt → $\$rd = \$rs * \$rt$

You should use the smallest number of instructions possible in each program. Compare the number of instructions and number of memory references (loads and stores) performed in each program.

This note relates to questions 3 and 4:

Use the spim simulator (available on the textbook's CD - please refer to page B-42 of the textbook for more information) to run and test your assembly program. Spim (and xspim) is a program that simulates the behavior of MIPS32 computers and can run MIPS32 assembly language programs. Documentation for spim is available in Appendix B of your textbook and at: <http://www.cs.wisc.edu/~larus/spim.html>. This spim website also contains a link for downloading a PC version of spim, if you'd

rather run it on your PC than on a dsil workstation. A helpful reference is provided for you at:

<http://www.ee.duke.edu/~sorin/ece152/resources/simple.s>.

This simple program sums the entries in a list of 9 integers.

3) [30] Write a MIPS assembly program to calculate the smallest number of coins necessary to add up to a specific amount of money, given in cents. The input will be a number ranging from 0 to 100. Assume that the available coin denominations are 1, 5, 10 and 25 cents. The output should be four numbers, following the format:

P N D Q

where P stands for the number of pennies (1c), N stands for the number of nickels (5c), D is the number of dimes (10c) and Q is the number of quarters (25c) in your solution. Notice that $P+N+D+Q$ should be minimal. Name your file *coins.s* and submit it on Sakai in the assignment tab.

4) [30] Write a MIPS assembly program that sorts, in ascending order, all of the characters given in an input string (sorting is done by ASCII code). You must use selection sort. Assume you will not get more than 100 characters. The pseudo code for the program is given below. You must use procedure calls and stack frames when implementing this code in assembly. (If you don't remember what insertion sort is, you can refresh yourself [here](#).) Name your file *sorting.s* and submit it on Sakai in the assignment tab.

```
main: while (read input char != carriage return)
    {
        add input char to array;
        //note that you should keep track of the number of
characters
    }

    for (each char in array)
    {
        call selectAndSort();
        //you must have this procedure in your assembly program
    }

    call printArray();
    //you must have this procedure in your assembly program
```