

Homework#2 for ECE 152

Instruction Sets (Chapter 2)

Due in class on Monday, February 11

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

- 1) [10 points] Patterson & Hennessy 2.30
- 2) [10 points] For an accumulator-based ISA, write (on paper) an assembly program that computes $X = A + (B * (C + D) + E)$. Assume that you have the following instructions: clear, add, multiply, where acc = “accumulator” and:

```
clear --> acc = 0
add A --> acc = acc + A
mult A --> acc = acc * A
```

- 3) [10 points] What MIPS assembly instruction is represented by 0x00544022?
- 4) [30] Write a MIPS assembly program that reads a string of integers and computes whether it (the number represented by the entire string) is prime or not. Assume the string of integers has no more than 5 digits. Use the spim simulator (available on the dsil computers) 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 A 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 a simple program that I've 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.

To submit your code for this question and the next question, create a directory in your ECE account. You can name this directory whatever you want, but for this explanation I'll assume it's named `~yourlogin/homework2`. Put your program files in this directory and name them `question4.s` and `question5.s`. Create a `.tar.gz` file called `homework2.tar.gz` that includes `question4.s` and `question5.s`, and then use the electronic submission website to upload `homework2.tar.gz`.

You may re-submit as often as you like, but a re-submission will overwrite whatever you've previously submitted for this assignment. I will grade whatever has been submitted before 10:00AM on Monday, February 11.

- 5) [30] Write a MIPS assembly program that sorts, in ascending order, all of the characters that are provided as inputs (sorting is done by ASCII code). Assume you won't get more than 100 characters. Some inputs may repeat. The pseudo-code for your program is below. Once again, use spim to run and test your code. Your program will be tested by the grader with several different arrays of characters, to make sure that your program does indeed work correctly.

```
main:    read one char;
         if (char == carriage return){
           exit program;
         } else {
           call insertAndSort(char);
         }
         call printArray();
         goto main;
// insertAndSort() must keep the array sorted in ascending order.
// You must have procedures for insertAndSort() and printArray().
// I want to see you use procedures and stack frames.
```