CSc 230 Fall 2015 - Midterm: Monday, October 19, 2015

What will be covered on the midterm?

- All class slides, numbered from 1 through 8.
- All labs from weeks 1 through 4.
- All material that was exercised in Assignments 1 and 2. (Although Assignment 2 programming will be submitted after the exam, so you can assume that the emphasis will be on Assignment 1 plus Assignment 2 written.)
- All material on the (posted) class reading list.

Midterm Exam Objectives

In order to successfully complete this midterm exam, you must be able to:

1. Define Computer Architecture and Computer Organization
    a. List and identify the various parts of a computer.
    b. Describe the internal parts of a CPU: ALU, registers, control unit, busses, input/output, etc.
    c. Explain the von Neuman and Harvard models of a Computer System. Compare /differentiate.
2. Count in decimal, hexadecimal, binary.
3. Convert between all combinations of decimal and hexadecimal and binary.
4. Represent negative integers in 2 ways, signed magnitude and 2's complement.
5. Perform arithmetic with each form of negative numbers.
6. Explain with arithmetic operations produce overflow. Identify overflow in results.
7. Explain why computer architectures would choose to use signed magnitude or 2's complement for negative integers.
8. Determine the range of integers available using a given integer representation and number of bits.
9. Explain how parity is used. Define even and odd parity and create parity bits.
10. Store numbers in and read from memory using little and big endian conventions.
11. Determine the size of busses based on the capacity of a system.
12. Perform logical operations (And, Or, invert) on binary values.
13. Shift and rotate binary values
14. Identify parts of the Fetch-Decode-Execute cycle.
    a. Describe the details of the Fetch-Decode-Execute cycle on a typical system.
15. Identify the opcode and operands of an AVR assembly language instruction
16. Determine the operation of an AVR assembly language program. (Ie, trace through a program and determine what is stored at the end of the execution.)
    a. Determine the values in the SVZNC bits of the status register after arithmetic or compare operations.
17. Distinguish between Absolute, Immediate and Direct addressing modes of an AVR assembly language program.

18. Complete an already started AVR assembly language program.
19. Write an entire AVR assembly language program that accesses data stored.
20. Be able to convert control structures such as if, if/else, switch, while, do/while and for into AVR assembly language.
21. Explain what an AVR 2560 stores in SRAM, Flash, and EEPROM.
22. What is the output of an assembler program?
23. What does a two-pass assembler produce on each of its two passes?
24. Be able to use common assembler directives.

A copy of the document instruction_set_summary.pdf will be available on each desk.