CS/ECE 252: INTRODUCTION TO COMPUTER ENGINEERING

UNIVERSITY OF WISCONSIN—MADISON

Prof. Guri Sohi

TAs: Rebecca Lam and Newsha Ardalani

Midterm Examination 3

In Class (50 minutes)

Friday, November 18, 2011

Weight: 17.5%

NO: BOOK(S), NOTE(S), CALCULATORS OF ANY SORT.

This exam has 10 pages, including one page for the LC3 Instruction Set and two blank pages at the end. Plan your time carefully, since some problems are longer than others. You must turn in pages 1 through 7.

LAST NAME: _____

FIRST NAME: _____

SECTION: _____

ID# _____

Problem	Maximum Points	Actual Points
1	2	
2	4	
3	6	
4	6	
5	6	
6	6	
Total	**30**	

Problem 1 (2 Points)

When a computer executes an instruction, the state of the computer is changed as a result of that execution. Is there any difference in the state of LC-3 computer as a result of executing instruction 1 below vs executing instruction 2 below? Explain. We can assume both instructions are located at the same address and the state of the LC-3 computer before execution is the same in both cases.

Inst 1 : 0001 000 000 1 00000 ; R0 <- R0 + #0

Inst 2: 0000 111 000000000 ; Branch to the next sequential instruction if any of P, Z or N is set

Problem 2 (4 Points)

A program wishes to load a value from memory into R1, and based on the value loaded into R1, executes code starting at x3040 if the loaded value is positive, executes code starting at x3080 if the value is negative, or executes code starting at location x3003 if the value loaded is zero. The first instruction of this program (load a value into R1) is shown at address x3000.

Your job: Write the instructions for locations x3001 and x3002, and comments for the instructions.

ADDRESS	INSTRUCTION	COMMENT
x3000	0010 001 011111111	Load a value from memory location x3100 into R1
x3001		
x3002		

Problem 3 (6 Points)

Answer the following short answer questions with no more than 4 sentences each.

a. (1 point) Suppose the number of opcodes for the LC-3 increases to 64. If the instruction size stays the same, how is the range of addresses that the BR instruction can branch to affected?

b. (1 point) Suppose the number of registers for the LC-3 is decreased by half. If the instruction size stays the same, how is the AND instruction (register mode) changed?

c. (2 points) Write two of the three constructs that comprise the systematic decomposition model and define them.

d. (2 points) Write two of the three different types of program errors and define them.

Problem 4 (6 Points)

An LC-3 program is located in memory locations x3000 through x3006. It starts executing at x3000. If we keep track of all values loaded into the MAR as the program executes, we will get a sequence that starts as follows. Such a sequence is referred to as a trace.

MAR Trace

x3000

x3005

x3001

x3002

x3006

x4001

x3003

x0021

We have shown below some of the bits stored in locations x3000 to x3006. Your job is to fill in each blank space with a 0 or 1, as appropriate.

x3000	0	0	1	0	0	0	0									
x3001	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1
x3002	1	0	1	1	0	0	0									
x3003																
x3004	1	1	1	1	0	0	0	0	0	0	1	0	0	1	0	1
x3005	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
x3006																

Problem 5 (6 Points)

Given the following LC-3 program, express the final value of R1 in terms of the initial value of R2 after execution of the last instruction. Show comments for each line. **No credit will be given without comments.**

ADDRESS	INSTRUCTION	COMMENT
x3000	0101 0110 1110 0000	
x3001	0001 0110 1110 0011	
x3002	0001 0010 1010 0000	
x3003	0001 0010 0100 0001	
x3004	0001 0110 1111 1111	
x3005	0000 0011 1111 1101	
x3006	1001 0100 1011 1111	
x3007	0001 0010 0100 0010	
x3008	1111 0000 0010 0101	

Problem 6 (6 points)

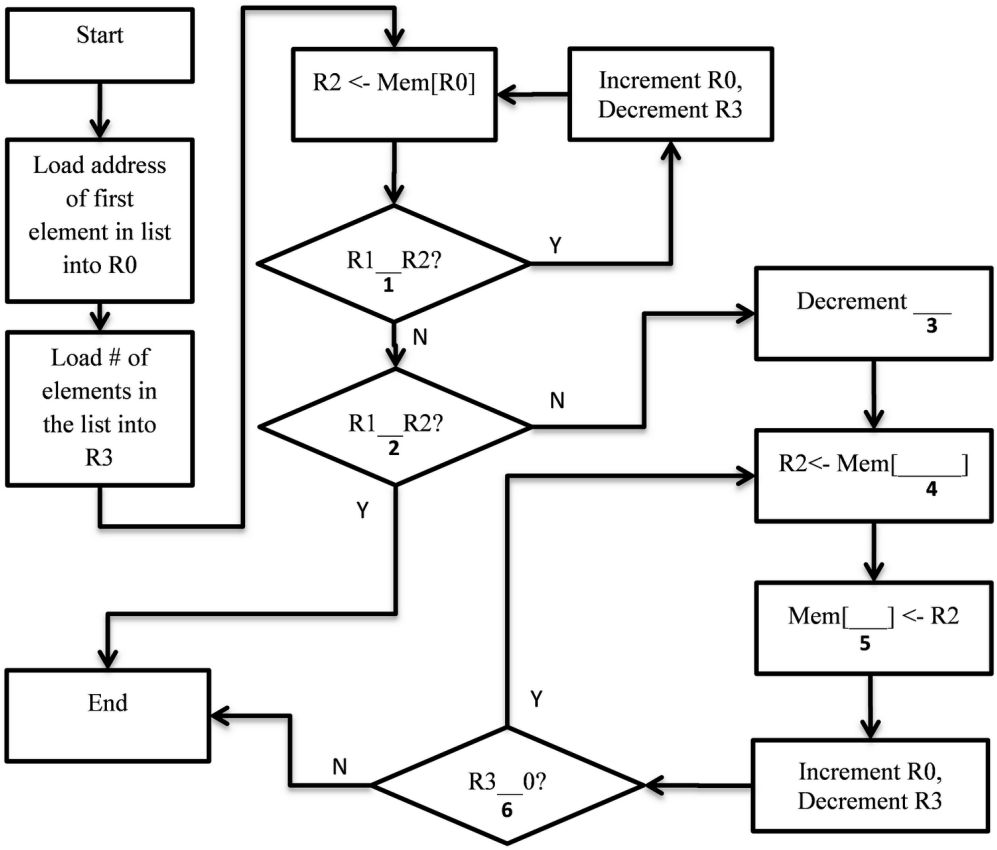
Suppose we wish to write a program that performs a deletion of one element from a list of elements sorted in ascending order without duplicates, where the element to be deleted is stored in R1. The total number of elements in the list is also known. The program works as follows:

Knowing that the element to be deleted is in R1, we load the first value of the list into R2. If the value of R2 is less than R1, we load the next value in the list into R2, and we keep doing this while R2 is less than R1. If the value of R2 is greater than R1, we halt the program. If the value of R2 is equal to R1, we load the next value in the list into the memory location that R2 used to be in. And we keep doing this for the rest of the list.

Example: The following table shows the list before and after x0003 is deleted from the list

Address	**Initial Value**	**Final Value**
x4500	x0001	x0001
x4501	x0002	x0002
x4502	x0003	x0004
x4503	x0004	x0005
x4504	x0005	x0006
x4505	x0006	unknown

Fill in the six missing blanks in the following flow chart. (Y = Yes, N = NO)



Extra Page 1

Extra Page 2

LC-3 Instruction Set (Entered by Mark D. Hill on 03/14/2007; last update 03/15/2007)

PC': incremented PC. setcc(): set condition codes N, Z, and P. mem[A]:memory contents at address A.
SEXT(immediate): sign-extend immediate to 16 bits. ZEXT(immediate): zero-extend immediate to 16 bits.