CS/ECE 252: INTRODUCTION TO COMPUTER ENGINEERING

UNIVERSITY OF WISCONSIN—MADISON

Prof. Karthikeyan Sankaralingam, Pradip Vallathol

TAs: Deepika Muthukumar, Sujith Surendran, Murali Sivalingam, Lisa Ossian

Midterm Examination 3

In Class (50 minutes)

Wednesday, April 09, 2014

Weight: 17.5%

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

The exam has **nine** pages. **Circle your final answers**. Plan your time carefully since some problems are longer than others. You **must turn in the pages 1-8**. Use the blank sides of the exam for scratch work. The Instruction set is provided on the last page.

LAST NAME: ____

FIRST NAME: ____

ID# ____

Problem 1 (6 points)

Please enter the missing values in the following LC-3 machine language program to implement a 2-input NOR function. Assume that the 2 inputs are stored in registers R1 and R2. The final output should be stored in register R3. (Adding comments to each machine language instruction will assist in awarding partial credit).

Instruction	Comment
1001 001 001 111111	R1 <- NOT(R1).
1001 010 010 111111	R2 <- NOT(R2).
0101 011 010 000001	R3 <- R1 (AND) R2.

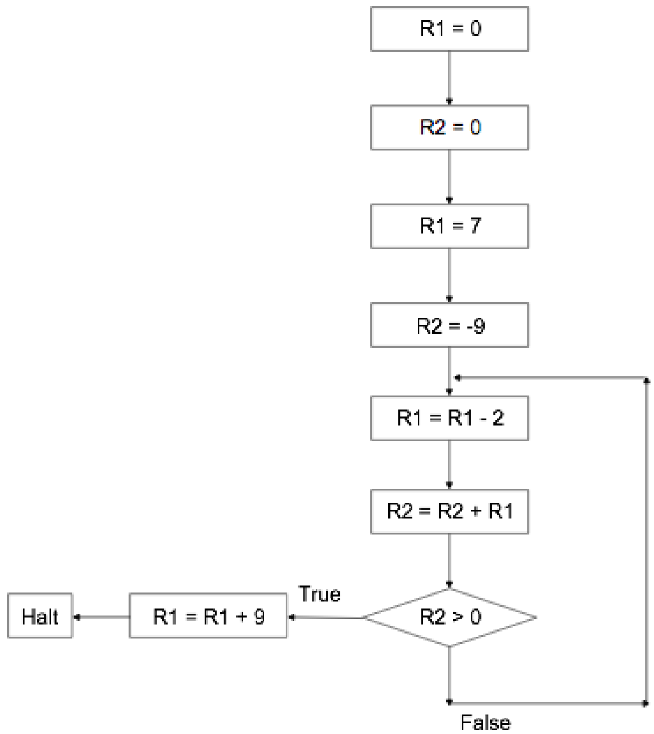
Problem 2 (3 points)

Please fill in the following LC-3 machine language program to multiply the value in R0 by the value in R1. Store the product of the multiplication in R2. (Adding comments to each machine language instruction will assist in awarding partial credit).

Note: The opcode bits [15:12] have been provided for each instruction.

Address	Instruction	Comment
x3200	0101 010 010 1 00000	R2 <- R2 (AND) 0.
x3201	0001 010 010 0 00 000	R2 <- R2 + R0.
x3202	0001 001 001 111111	R1 <- R1 - 1.
x3203	0000 0 0 1 111111101	BR x3201.
x3204	1111 0000 00100101	Halt.

Problem 3 (4 points)



Consider the program represented by the flowchart above.
Please complete the table (2 instructions and 6 comments) in accordance with the flow chart.

Address	Instruction	Comment
x3000	0101 001 001 1 00000	Clear R1.
x3001	0101 010 010 1 00000	Clear R2.
x3002	0001 001 001 1 00111	Set R1 to 7.
x3003	0001 010 010 1 10111	Set R2 to -9.
x3004	0001 001 001 1 11110	R1 = R1 - 2.
x3005	0001 010 010 0 00 001	R2 = R2 + R1.
x3006	0000 110 111111101	BRnz 0x3004.
x3007	0001 001 001 1 01001	R1 = R1 + 9.

x3008	1111 0000 00100101	HALT

Problem 4 (4 points)

This problem has 2 parts which consider the following LC-3 instruction located at memory address x4300.

```
x4300: 0000 010 000000100
```

a. Assume the initial contents of the registers are as follows:

R0 contains 4
R1 contains 3
R2 contains 6
R3 contains 0

If the instruction at 0x42FF is the one shown below, what is the value of the PC after the instruction at 0x4300 is executed?

```
x42FF: 0001 000 001 1 00001
x4300: 0000 010 000000100
```

x4301

b. Assume the initial contents of the registers are as follows:

R0 contains 4
R1 contains 3
R2 contains 6
R3 contains 0

If the instruction at 0x42FF is the one shown below, what is the value of the PC after the instruction at 0x4300 is executed?

```
x42FF: 0101 011 001 1 00000
x4300: 0000 010 000000100
```

x4305

Problem 5 (5 points)

The PC has the value `x5010`. The following memory locations contain values as shown:

Memory Location	Contents
x5013	x6023
x6023	x6025
x6025	x0112
x6027	x824C

a. (3 points) The following LC-3 instructions are then executed, causing a value to be loaded into R3. What is that value?

```
x5010: 1110 1000 0000 0010
x5011: 0110 1011 0000 0000
x5012: 0110 0111 0100 0000
```

LEA R4, x5013; R4 <- x5013
LDR R5, R4, x0000; R5 <- Mem[x5013] = x6023
LDR R3, R5, x0000; R3 <- Mem[x6023] = x6025

b. (1 point) In English words, describe what the program (the three-instructions sequence) is accomplishing.

LDI

c. (1 points) We could replace the three-instruction sequence (in part a) with a single instruction (at memory location `x5010`). What is it ? (Show the 16 bits of the instruction)

`1010 011 000000010`

Problem 6 (2 points)

Which of the following LC-3 instructions at address `0x0350` will always clear register R0 (i.e. set the contents of R0 to all zeroes) ?

```
a. 0001 000 000 100000
b. 0101 000 000 100000
c. 1110 000 000 000000
d. 0010 000 000 000000
```

b.

Problem 7 (3 points)

a. (1 point) In any program, what is the range of values that the immediate field of an ADD instruction can take?

-2^4 to 2^4-1.

b. (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 a BR instruction can access affected?

Opcode field is increased by 2 bits. PC offset becomes 7 bits and PC offset range becomes -2^6 to 2^6-1. Thus the branch can be from $PC+1-2^6$ to $PC+2^6$

c. (1 point) An LDR instruction, located at x3000, uses R1 as its base register. The value currently in R1 is x1000. What is the largest address that this instruction can load from?

`x1000 + x001F = x101F.`

Problem 8 (3 points)
In this problem, the program starts executing from address `0x3000`.

If the value stored in R0 is 0 at the end of the execution of the instruction at `0x3003`, what can be inferred about R5?

```
0x3000: 0101 000 000 1 00000
0x3001: 0101 100 101 1 000001
0x3002: 0000 010 000000001
0x3003: 0001 000 000 1 00001
```

a. R5 is equal to 0
b. R5 is equal to 1
c. R5 is equal to 2
d. R5 is equal to 3

Hint: Start with the last instruction and work backwards. If there is a branch instruction, decide if that branch is taken or not, for the value in R0 to be 0.

a.