CS/ECE 252: INTRODUCTION TO COMPUTER ENGINEERING

UNIVERSITY OF WISCONSIN—MADISON

Instructor: Rahul Nayar

TAs: Annie Lin, Mohit Verma

Examination 3

In Class (50 minutes)

Wednesday, April 7, 2017

Weight: 17.5%

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

The exam has ten pages. You must turn in the pages 1-9. Circle your final answers. Plan your time carefully since some problems are longer than others. Use the blank sides of the exam for scratch work.

LAST NAME: _____

FIRST NAME: _____

ID#: _____

Problem	Maximum Points	Points Earned
1	3	
2	3	
3	4	
4	5	
4	5	
6	3	
7	3	
Total	26	

Problem 1

Address	Instruction	Comment
0x3000	0010 000 000101010	R0 <- M[0x302B]
0x3001	0010 001 000101010	R1 <- M[0x302C]
0x3002	1001 011 001 111111	R3 <- NOT (R1)
0x3003	0001 011 011 1 00001	R3 <- R3 + 1
0x3004	0001 010 000 0 00 011	R2 <- R0 + R3
0x3005	0000 100 000000010	BRn 0x3008
0x3006	0011 000 000101000	M[0x302F]<- R0
0x3007	0000 111 000000001	BRnzp 0x3009
0x3008	0011 001 000100110	M[0x302F]<- R1
0x3009	1111 0000 0010 0101	HALT

Problem 2

Given the initial values at the following registers and memory locations, fill in the values at the memory locations below after each instruction is executed. The instructions are executed **in order**. So, instruction at location x4000 has finished execution before instruction at x4001 begins, and so on. You may assume that all other registers and memory locations are set to 0.

Address	Initial Memory Values
R0	x4022
R1	x4023
R2	x4024
x4020	x4022
x4021	x4023
x4022	xFFFD

Address	LC-3 Binary Instruction	Values at memory locations after execution
x4000	0110 010 000 000000 LDR R2, R0, 0	Value at x4020: x4022 Value at x4021: x4023

	(R2 = xFFFD)	Value at x4022: xFFFD
x4001	0001 001 010 0 00 001 ADD R1, R2, R1 (R1 = x4020)	Value at x4020: x4022 Value at x4021: x4023 Value at x4022: xFFFD
x4002	0111 010 001 000000 STR R2, R1, 0	Value at x4020: xFFFD Value at x4021: x4023 Value at x4022: xFFFD

Problem 3

The following pseudo-code presents an algorithm to check if the data present in R1 is greater than 4. The table below shows an incomplete LC-3 binary program that implements this logic. **Assume that R1 has been initialized to the data value being checked.**

```
R1 = R1 - 4
R2 = 0
If R1 > 0 then:
        R2 = 1
end if
HALT
```

Assume PC is x3000 when execution of the program starts.

Address	Instruction
0x3000	0101 010 010 1 00000 R2 = 0
0x3001	0001 001 001 1 11100 ADD R1, R1, -4
0x3002	0000 110 000000001 BRnz x3004
0x3003	0001 010 010 1 00001 ADD R2, R2, 1
0x3004	1111 0000 00100101 HALT

a) Complete the code to implement the algorithm in the above table by filling in the missing LC-3 binary instructions in memory locations 0x3001 and 0x3002.

b) By looking at the algorithm logic above, a student incorrectly concludes that if R2 = 1 at the **end** of program execution, the value in R1 at program **start** must be **greater than** 4. Provide at least one example of a value in R1 for which the above conclusion is incorrect.

Fails for all values in R1 which are large negative, and would cause overflow while calculating R1 - 4. Eg. R1 = -2^15 or 0x8000

c) Which of the following programming construct does the above algorithm use?

i) Iterative

ii) Conditional

ii) Conditional

Problem 4

a) Briefly explain what the following LC-3 instruction does:

```
0000 111 000 0 00000
```

Unconditionally branches to the immediately next instruction. In other words, a NO-OP or does nothing.

b) Which of the following instructions **does not** change the condition code of LC-3 after execution? **You must explain your answer for full credit.**

```
a) 0101 011 000 000 001
b) 0110 011 010 000011
c) 1011 011 010000011
d) 1010 110 000000011
```

c) as it is a STI instruction, it doesn't modify any register and doesn't change condition codes.

c) How many memory accesses does the LDI instruction in LC-3 ISA make? **You must explain your answer for full credit.**

2

d) Briefly explain the difference between syntax errors and logical errors.

Syntax errors are language errors that can be caught by the compiler. Logic errors are errors that are caught in execution or lead to an incorrect result.

e) The following instruction is located in memory at 0x3000.

```
0000 111 000001111
```

What is the value of PC after the instruction finishes execution? Assume n = 1, z = 0, p = 0 before the instruction begins execution.

PC = 0x3010

Problem 5

The following table shows an incomplete program located in memory. Assume PC = x3000 before the program starts execution.

Address	Instruction	Comments
0x3000	1001 001 001 111111	R1 <- NOT (R1)
0x3001	1001 010 010 111111	R2 <- NOT (R2)
0x3002	0101 001 001 0 010	R3 <- R1 AND R2
0x3003	1001 011 011 1 11111	R3 <- NOT R3

0x3004	1011 011 000000001	M[x4000] <- R3
0x3005	1111 0000 00100101	HALT
0x3006	0100 0000 0000 0000	.FILL x4000

a) Fill in the missing LC-3 binary instructions from the comments provided.

b) The following table shows the values in select registers and condition flags before the execution of the above program begins. Write the values in these locations just **after** the program finishes execution (i.e. after HALT has finished execution).

Register/Condition flag	Value before execution starts	Value after execution completes
R1	0x00FB	0xFF04
R2	0x00FA	0xFF05
R3	0x0000	0x00FB
n	1	0
p	0	1
z	0	0
M[x4000]	0x0000	0x00FB

Problem 6

a) Write a **single** LC-3 instruction to load the number x3010 into R2. Your instruction will be located at x3000.

1110 010 000001111 LEA R2, 15 -> xE40F

b) Write a **single** LC-3 instruction to store the data from R4 into memory address x3FF0. Your instruction will be located at x4000.

0011 100 111101111 ST R4, -17 -> x39EF

c) Write **up to two** LC-3 instructions that will subtract the number 30 from R3 and place the result in R2.

0001 011 011 1 10001 ADD R3, R3, -15
0001 010 011 1 10001 ADD R2, R3, -15

Problem 7

	Before	After
R0	xFF35	xFF35
R1	xF911	xF911
R2	x67F9	x0146
R3	x0912	x0912
R4	x8231	x8231
R5	x3040	x3040
R6	x0901	x0901
R7	x11AA	x11AA
x304D	x1091	x1091
x304E	x7684	x7684
x304F	x9A2F	x9A2F
x3050	xEFFF	xEFFF
x3051	x0146	x0146
x3052	x99DF	x99DF
x3053	x4782	x4782
x3054	xA221	xA221

Instruction:

LC-3 Binary Form	Comment
0110 010 101 010001	LDR R2, R5, 17

Explanation:

The value that changes is R2, which is loaded from x3051. x2000 is too far to load via LD, and there are no instructions that contain a memory address relevant to LDI. R5 contains x3040 which is close enough to x3051 to load with an offset of 17.