York University
Lassonde School of Engineering
Dept. of Electrical Engineering and Computer Science
Fall 2015

EECS2021	**Midterm**	**Computer Organization**
Wednesday, Oct. 21st, 2015		5:30 – 6:45

Last Name |__|__|__|__|__|__|__|__|__|__|__| First name| _|__|__|__|__|__|__|__|__|__|__|

ID ______________________

Solution

B

Instructions to students:

Answer all questions.
Marks are shown in front of each question number.
Show your work
Be neat and clean while drawing your logic, block, or state diagrams.

This examination consists of **FIVE** questions

Problem	Points	problem	Points
1	/8	2	/6
3	/4	4	/5
5	/8	6	/4
Total	**/35**		

Problem 1 (8 points)

Answer each of the following questions:

(a) What is the minimum (most negative) two's complement number that can be represented in 5 bits?

10000 = −16

(b) What is the maximum two's complement number that can be represented in 5 bits?

01111 = 15

(c) What does 10101001 represent in unsigned format?

$2^0 + 2^3 + 2^5 + 2^7 = 169$

(d) What does 10101001 represent in 2's complement format

$2^0 + 2^3 + 2^5 - 2^7 = -87$

OR

take 2's Comp

10101001
01010110
       1
01010111 = −87

(e) Represent the number +7 in 4 bits using 2's complement format

7 : 0111

(f) represent the number -6 in 4 bits using two's complement format

-6 :

0110
1001
1
1010

(g) Add the numbers in (e) and (f) using 2's complement, show the actual addition and what is the result?

add

0111
1010
0001 = 1

Problem 2 (6 points)

Instruction	Cycles	percentage
load	6	20%
store	3	20%
add/sub	1	30%
multiply	10	10%
branch	3	20%

a) What is the average CPI?

1pt

6(0.2) + 3(0.2) + 1(0.3) + 10(0.1) + 3(0.2)

= 3.7

b) A program executes 10 billion instructions, and a clock rate of 300MHz, how long to execute this program on the above machine?

2pt

$\frac{10 \times 10^9 \times 3.7}{300 \times 10^6}$ = 123.33 sec

OR 10^9(10.2×6+ ....)

c) In order to improve the design, you have two choices; the first is to reduce multiplication time to 9 cycles. The second is change load time to 1 cycle, but that means you have to increase multiply to 11. Which one is faster? What is the increase in the performance for the fastest solution compared to the original CPU?

3pt.

Case I

6(0.2) + 3(0.2) + 1(0.3) + 9(0.1) + 3(0.2)

= 3.6

Case II

1(0.2) + 3(0.2) + 1(0.3) + 11(0.1) + 3(0.2)

= 2.8

Case II faster by ~~3.6/2.8 = 1.32 times~~

$\frac{3.7}{3.6}$ = 1.03 times

3%

Problem 3 (4 points)

A given application is written in Java runs 10 seconds on a desktop processor. A new Java compiler is released that requires only 70% as many instructions as the old compiler. Unfortunately, it increases the CPI . We found out that the execution time of the new machine is 12 seconds. What is the new CPI compared to the old one.

$T_{before} = IC \times CPI \times T_c$ 2 pt.

$T_{after} = 0.7\ IC \times CPI_a \times T_c$

T_c = 1 / clock rate (clock time); T_{after} = execution time; IC = # of instructions; clock cycles = execution time / clock time

$$\frac{T_b}{T_a} = \frac{10}{12} = \frac{IC \times CPI \times T_c}{0.7 I_c \times CPI_a \times T_c}$$

(T_a: execution time)

2 pt.

$7\ CPI_a = 12\ CPI$

$CPI_a = \frac{12}{7}$ times CPI before

↳ 1.714

Problem 4 (5 points)

Consider the following piece of code.

```
addi  $s0, $zero, 4
addi  $v0, $zero, -2
add   $s1, $s0, $v0
beq   $s1, $v0, Skip
sll   $s2, $s1, 1
Skip
or $v0, $v0, $s2
```

Register $s0 = 4

Register $v0 = -2

Register $s1 = 2

Register $s2 = 4

1pt. each +1

Problem 5 (8 points)

Given the following code

Memory Address	Data Value
0x23180000	0x00000000
0x23180004	0x00000010
0x23180008	0x00001000
0x2318000C	0x00010000
0x23180010	0x00100000

Write the contents of the registers mentioned after the '#' **after the execution of the instruction to the left**

1 each

Instruction	Register
`main:`	
`lui $a0, 0x2318`	**#$a0=** 0x23180000
`ori $a0, $a0, 0x0004`	**#$a0=** 0x23180004
`lw $t0, 8($a0)`	**#t0=** 0x00010000
`addi $sp, $sp, -4`	
`sw $t0, 0($sp)`	**#$t0=** 0x00010000
`lw $t1, 0($a0)`	**#$t1=** 0x00000010
`or $t3, $t0, $t1`	**#$t3=** 0x00010010
`sw $t3, 4($a0)`	**#$t3=** 0x00010010
`addi $sp, $sp, -4`	
`sw $t1, 0($sp)`	
`lw $t3, 0($sp)`	**#$t3=** 0000 0010
`addi $sp, $sp, 4`	
`lw $t4, 0($sp)`	**#$t4=** 0001 0000

stack
0001 0000
0000 0010

Fill in the contents of the memory after the completion of the above code

Memory Address	Data Value
0x23180000	0x 0
0x23180004	0x 010
0x23180008	0x 10010
0x2318000C	0x00010000
0x23180010	0x00100000

→ only one changed → 1

no change 1

Problem 6 (4 points)

- In MIPS, there are many branch instructions `beq` `bne, br, jr`
  Which of these branches has the largest reach (can jump further away than any other branch)? Briefly explain

Jr because it uses a 32-bit address in a register

- There are many instruction formats in MIPS, draw 2 instruction format and state what they are used to (one example each).

6	5	5	5	5	6
op	r	r	r	shamt	fun

add $t1, $t2, $t3 or just add

6	5	5	16
op	r	r	Imm