

Name: KEY

Id#

## COE 301/ICS 233, Term 151

### Computer Architecture & Assembly Language

#### Quiz# 2

Date: Tuesday, Oct. 6, 2015

**Q1.** Fill the blanks in the following questions:

(1) Assume that the instruction `j NEXT` is at address `0x00400010` in the text segment, and the label `NEXT` is at address `0x00400fec`. Then, the address stored in the assembled instruction for the label `NEXT` is  $0x00400fec/4=0x01003fb$

(2) Assume that the instruction `bne $t0, $t1, NEXT` is at address `0x00400010` in the text segment, and the label `NEXT` is at address `0x00400fec`. Then, the address stored in the assembled instruction for the label `NEXT` is  $(0x00400fec-0x00400014)/4=0xfd8/4=0x03f6$

(3) The pseudo instruction `ori $t0, 0x12345678` is implemented by the following minimum MIPS instructions:

```
lui $at, 0x1234
ori $at, $at, 0x5678
or $t0, $t0, $at
```

(4) After executing the instruction `addu $t0, $s1, $s2`, the following MIPS instruction can be used to store the carry out of addition in register `$t1`:

```
sltu $t1, $t0, $s1
```

- (5) To multiply the **signed** content of register \$t0 by 62.5 without using multiplications and division instructions, we use the following MIPS instructions:

```
sll $t1, $t0, 6
sra $t2, $t0, 1
subu $t1, $t1, $t2
subu $t0, $t1, $t0
```

- (6) The content of register \$s1 after executing the following code is 0x4.

```
li $s0, 0x5a
li $s1, 0
Next:
andi $t0, $s0, 1
add $s1, $s1, $t0
srl $s0, $s0, 1
bne $s0, $0, Next
```

- (7) The code given below implements the conditional statement:

```
if (($t0 < 1) OR ($t1 > 100)) then $t2=0
```

```
slti $t3, $t0, 1
bne $t3, $zero, Zero_index
li $t3, 100
slt $t3, $t3, $t1
beq $t3, $zero, End_if
Zero_index:
xor $t2, $t2, $t2
End_if:
```