# COE 301: Computer Organization – Term 211

## Quiz 2: MIPS ALU Instructions, Pseudo-Instructions, and Control Flow

## SOLUTION

1. (4 pts) Rewrite the following pseudo-instructions using one or more equivalent basic MIPS instructions. You may use ONLY the **$at** register as a temporary register for intermediate results. Notice that **0x1234ABCD** is a 32-bit constant that does not fit in the I-type format. **ROL** (Rotate Left) is a pseudo-instruction that shifts the bits to the left, such that the **upper 5 bits** of **$t4** are rotated to become the **lower 5 bits** of **$t0**.

| Pseudo-Instruction | Equivalent Basic MIPS Instructions |
|---|---|
| XORI  $t1, $t2, 0x1234ABCD | LUI  $at, 0x1234<br>ORI  $at, $at, 0xABCD<br>XOR  $t1, $t2, $at |
| ROL   $t0, $t4, 5 | SLL  $at, $t4, 5<br>SRL  $t0, $t4, 27<br>OR   $t0, $t0, $at |

2. (6 pts) Translate the following nested if-statements into MIPS assembly code. All the variables are signed. The values of **a**, **b**, and **c** are stored in **$t0**, **$t1**, and **$t2**, respectively.

```
if ((a >= 0) && (a <= 9)) {
  if (b>c) a = a * 10;
  else a = a / 4;
}
```

```
      BLTZ   $t0, skip       # if (a < 0) skip if
      LI     $t3, 9          # t3 = 9
      BGT    $t0, $t3, skip  # if (a > 9) skip if
      BLE    $t1, $t2, else  # if (b <= c) branch to else
      SLL    $t3, $t0, 1     # $t3 = a * 2
      SLL    $t4, $t0, 3     # $t4 = a * 8
      ADDU   $t0, $t3, $t4   # a = a * 10
      J      skip            # skip else
else: SRA    $t0, $t0, 2     # a = a / 4
skip:
```