



# CptS260: Introduction to Computer Architecture

Fall 2020

## Homework 3

**Due: Friday 10/02/2020 at 11.59 pm**  
**School of Electrical and Computer Engineering**

*Note: Submit your answers as an electronic copy of your answers through Blackboard.*

### Part 1: Intro to Assembly

1. Translate the following MIPS code into C. Assume that the variables  $f, g, h, i$ , and  $j$  are assigned to registers  $\$s0, \$s1, \$s2, \$s3$ , and  $\$s4$ , respectively. Also assume that the base address for arrays A, and B are stored in  $\$s6, \$s7$ , respectively.

```
sll $t0, $s0, 2
add $t0, $s6, $t0
sll $t1, $s1, 2
add $t1, $s7, $t1
lw $s0, 0($t0)
addi $t2, $t0, 4
lw $t0, 0($t2)
add $t0, $t0, $s0
sw $t0, 0($t1)
```

2. Assume that registers  $\$s0$ , and  $\$s1$  hold the value  $0x80000000$  and  $0xD0000000$ , respectively. (0x: means a hexadecimal number)
  - a. What is the value of  $\$t0$  after execution of the following instruction? Do we have overflow here? Explain your reasoning.

```
add $t0, $s0, $s1
```

- b. What is the value of  $\$t0$  after execution of the following instruction? Do we have overflow here? Explain your reasoning.

```
sub $t0, $s0, $s1
```

- c. What is the value of \$t0 after execution of the following assembly code? Do we have overflow here? Explain your reasoning.

```
add $t0, $s0, $s1
add $t0, $s0, $t0
```

3. For the following loop, write the equivalent C code routine. Assume that the registers \$s1, \$s2, \$t1, and \$t2 are integers A, B, i, and temp, respectively.

*LOOP:*

```
slt $t2, $0, $t1
beq $t2, $0, DONE
subi $t1, $t1, 1
addi $s2, $s2, 2
j LOOP
```

*DONE:*

## Part 2: Intro to MIPS

1.

- a. Provide the type and assembly language instruction for the following binary value:

$(0010\ 0001\ 0010\ 1001\ 0000\ 0000\ 0000\ 0001)_{(2)}$

- b. Provide the type and hexadecimal representation of following instruction:

`sw $t5, 32($t2)`

- c. Provide the type, assembly language instruction, and binary representation of instruction described by the following MIPS fields (all numbers are represented in decimal):

op=0, rs=3, rt=2, rd=13, shamt=0, funct=36

2. The following instruction is not included in the MIPS instruction set:

`rpt $t2, loop`       $\# \text{if}(R[rs] > 0) R[rs] = R[rs] - 1, PC = PC + 4 + \text{BranchAddr}$

- a. If this instruction were to be implemented in the MIPS instruction set, what is the most appropriate instruction format
- b. What is the shortest sequence of MIPS instructions that performs the same operation?
3. Show the compiled MIPS code for the following C code, assume x and y is in \$s1 and \$s2 respectively, and the base address of A is in \$s3.

$$y = x + A[5] + A[4]$$

$$A[6] = A[6] - y$$