

CMPEN331 – Quiz 2 (February 8, 2022)

Section:

Major:

Name:

Email:

- (3 points) For each MIPS instruction, show the value of the source register (RS), and target register (RT) fields. For the I-type instructions, show the value of the immediate field, and for the R-type instructions, show the value of the destination register (RD) field.

	type	rs	rt	rd	immed
addi \$t0, \$s6, 4					
add \$t1, \$s6, \$0					
sw \$t1, 0(\$t0)					
lw \$t0, 0(\$t0)					
add \$s0, \$t1, \$t0					

Given the following:

Register Name	Register Number	Usage
\$zero	0	Constant 0
\$at	1	Reserved for assembler
\$v0, \$v1	2, 3	Function return values
\$a0 - \$a3	4 - 7	Function argument values
\$t0 - \$t7	8 - 15	Temporary (caller saved)
\$s0 - \$s7	16 - 23	Temporary (callee saved)
\$t8, \$t9	24, 25	Temporary (caller saved)
\$k0, \$k1	26, 27	Reserved for OS Kernel
\$gp	28	Pointer to Global Area
\$sp	29	Stack Pointer
\$fp	30	Frame Pointer
\$ra	31	Return Address

	type	rs	rt	rd	immed
addi \$t0, \$s6, 4	I-type	22	8		4
add \$t1, \$s6, \$0	R-type	22	0	9	
sw \$t1, 0(\$t0)	I-type	8	9		0
lw \$t0, 0(\$t0)	I-type	8	8		0
add \$s0, \$t1, \$t0	R-type	9	8	16	

- Call a function with one argument. To be specific, translate `b = Func(17);` assuming `b` is to be placed in register `$s1`. You have to manually assign the argument. No more than 3 instructions. (4 points)

```
li    $a0, 17    or    addi $a0, $zero, 17
jal   Func       jal   Func
move  $s1, $v0   or    add  $s1, $v0, $zero
```

- (3 points) Translate the following C code to MIPS. Assume that the variables `f`, `g`, `h`, `i`, and `j` are assigned to registers `$s0`, `$s1`, `$s2`, `$s3`, and `$s4`, respectively. Assume that the base address of the arrays `A` and `B` are in registers `$s6` and `$s7`, respectively. Assume that the elements of the arrays `A` and `B` are 4-byte words:

`B[g] = A[i] + A[j];`

```
sll t0, s3, 2  # $t0 = 4*i
add t0, s6, t0 # $t0 = Base address of A + 4*i
```

```
sll t1, s4, 2    # $t1 = 4*j
add t1, s6, t1   # $t1 = Base address of A + 4*j
lw t0, 0(t0)    # t0 = the value of A[i]
lw t1, 0(t1)    # t1 = the value of A[j]
add t2, t0, t1  # t2 = A[i] + A[j]
sll t0, s1, 2   # $t0 = 4*g
add t0, s7, t0  # t0 = Base address of B + 4*g
sw t2, 0(t0)    # t2= value of B[g]
```