

Craig Shaffer

Dr. Swami Ponponi

CPSC 300

3-13-2022

Homework 8

1. Write MIPS code for the below Python code. [20]

Addresses of i @ $0xF0000000$, j @ $0xC004F004$, base address of A is $0xF0000004$

While $(i+j) \geq 10$:

$A[i] = A[i] + A[j]$

$i = i + 1$

$j = j + 2$

MIPS code:

```
li $t0, 0xF0000000 # load i into $t0
ori $t0, 0x00000000
li $t1, 0($t0)
li $s0, 0xC004F004 # load j into $s1
ori $s0, 0xF0000000
li $s1, 0($s0)
li $s2, 0xF0000004 # $s2 has base address of A
ori $s2, 0x00000000
addi $s3, $zero, 10 # 10 -> $s3
L2: add $s4, $t1, $s1 # (i+j) -> $s4
bne $s4, $s3, L2 # if (i+j) != 10 jump to L2
sll $s5, $t1, 2 # i*4 -> $s5
add $s5, $s5, $s2 # absolute address of A[i] -> $s5
lw $s6, 0($s5) # load A[i] to $s6
sll $s7, $s1, 2 # j*4 -> $s7
add $s7, $s7, $s2 # absolute address of A[j] -> $s7
lw $t2, 0($s7) # A[j] -> $t2
add $t2, $s6, $t2 # A[i] + A[j] -> $t2
sw $t2, 0($s5) # (A[i] + A[j]) -> $t2 -> A[i]
addi $t1, $t1, 1 # i = i + 1
addi $s1, $s1, 2 # j = j + 2
j L2 # jump to L2
L1: ←
```

2. In the MIPS simulation code, add the following modifications to mips_regfile.py. [20]

a. Complete the Num2Reg function to return the alphanumeric register name (\$t0, \$s0 etc.) given the register number (8, 16 etc.).

Dictionary used on line 28 and the completed Num2Reg function is on line 41 in mips_regfile.py

b. \$zero register always reads 0 in MIPS and data cannot be written into \$zero register. Modify the code to reflect this fact.

Modification on line 60-62 in mips_regfile.py. I added an if statement to check if the destination register is equal to 0 (\$zero). If it is, it won't write into the register. If it's not equal to 0, then it will write the data into the destination register like normal.

Test code is located at line 94 in mips_regfile.py