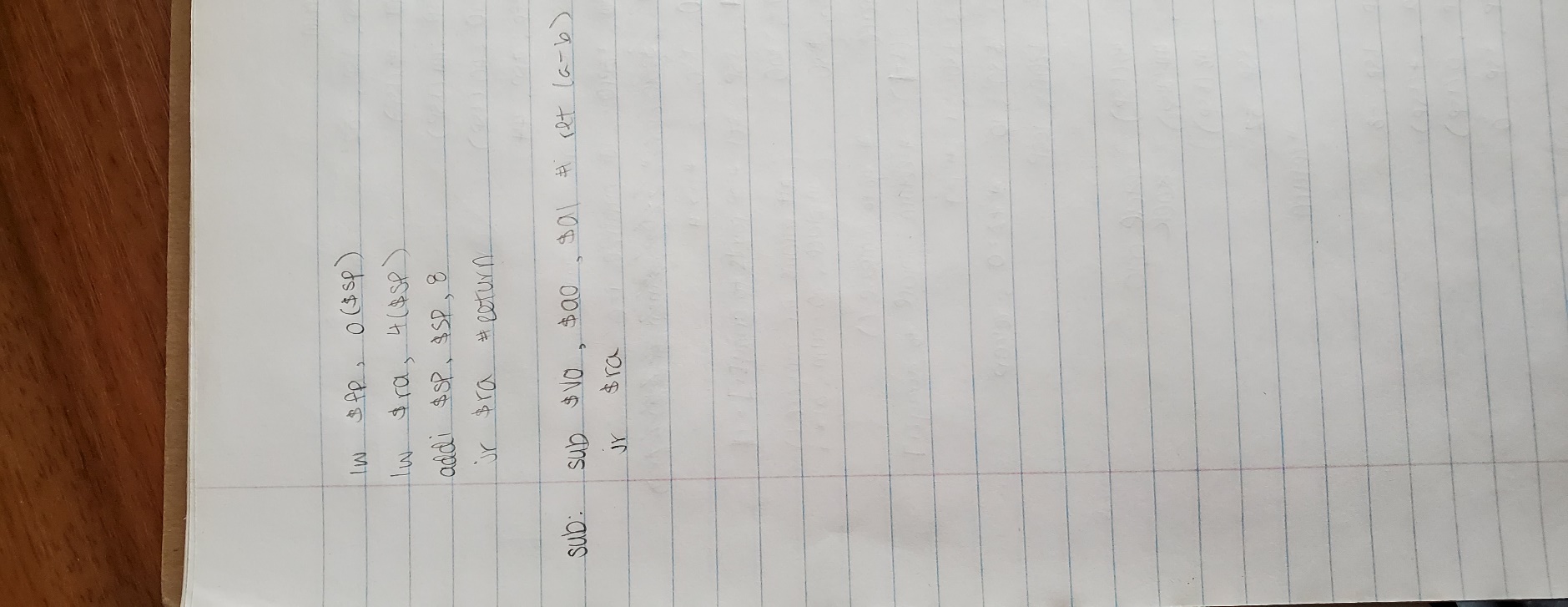
Anna Tang

3180300155

**Computer Organization: Homework 2**

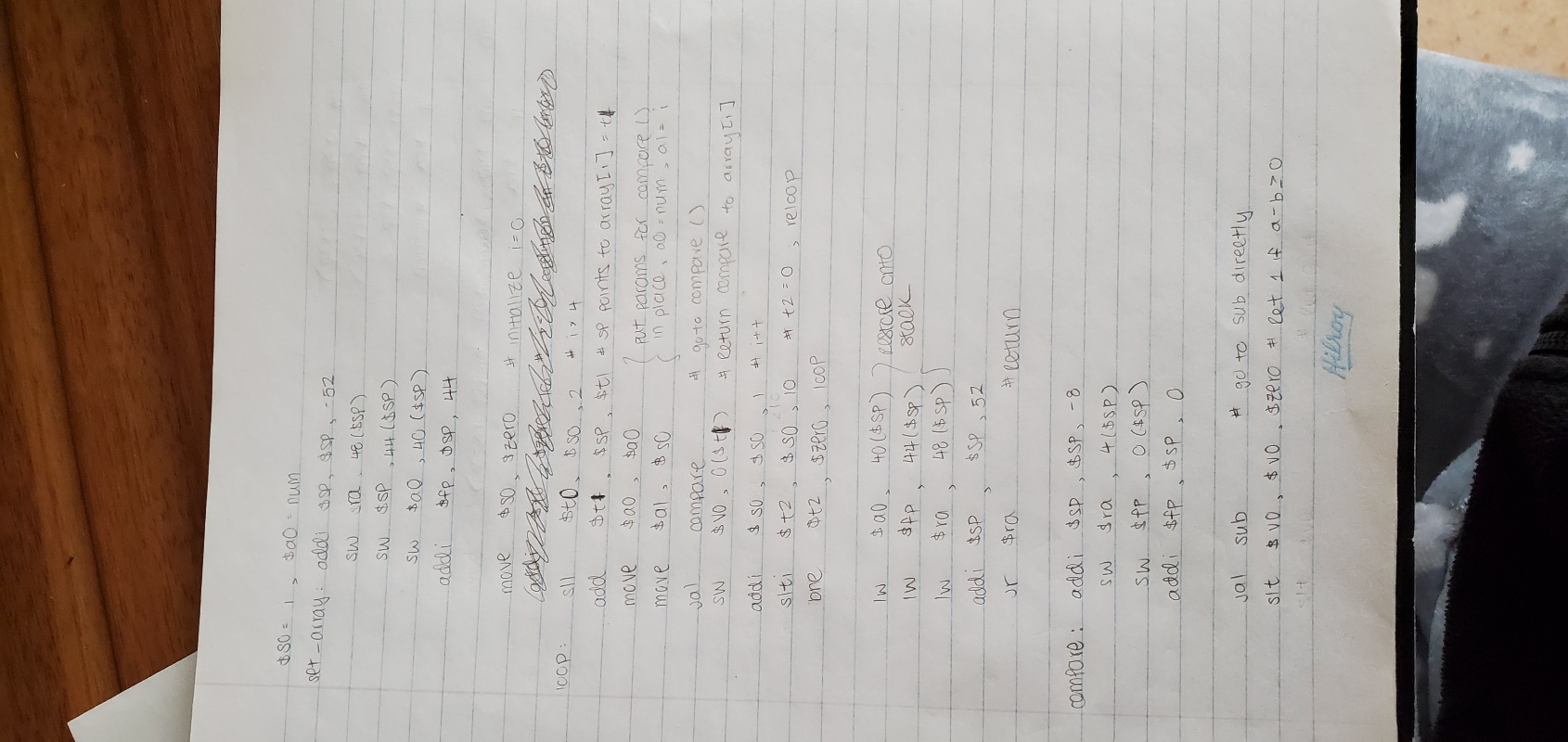
2.4) MIPS doesn’t have a subtract immediate instruction because it already has an add immediate instruction (addi). One could simply add an negative number and that would suffice, making a subi instruction unnecessary.

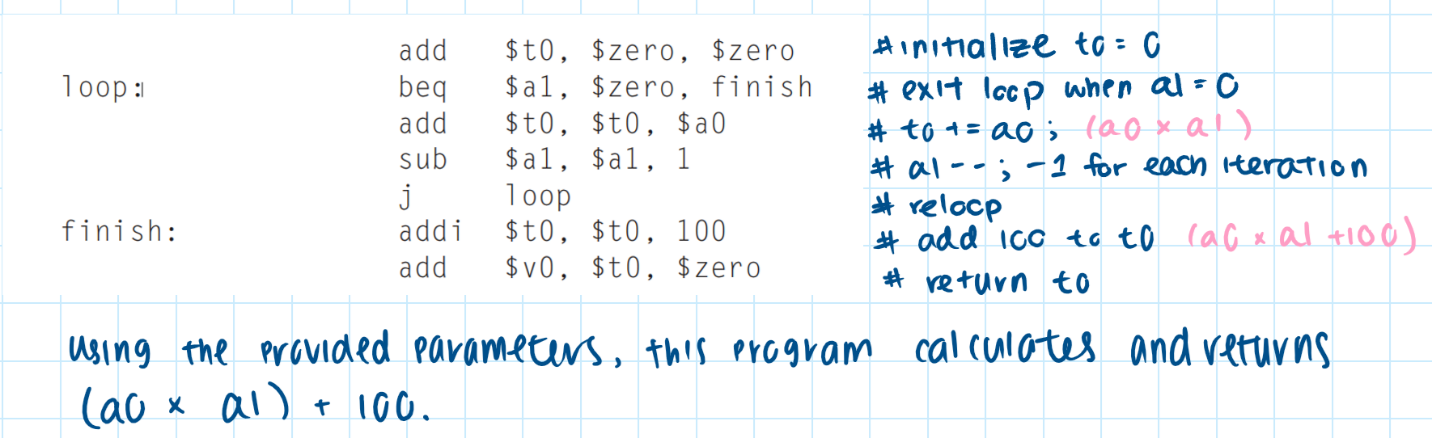
2.6) To extract an arbitrary field from a 32-bit register from $t3 to $t0, we can use the following instructions:

sll $t0, $t3, 9

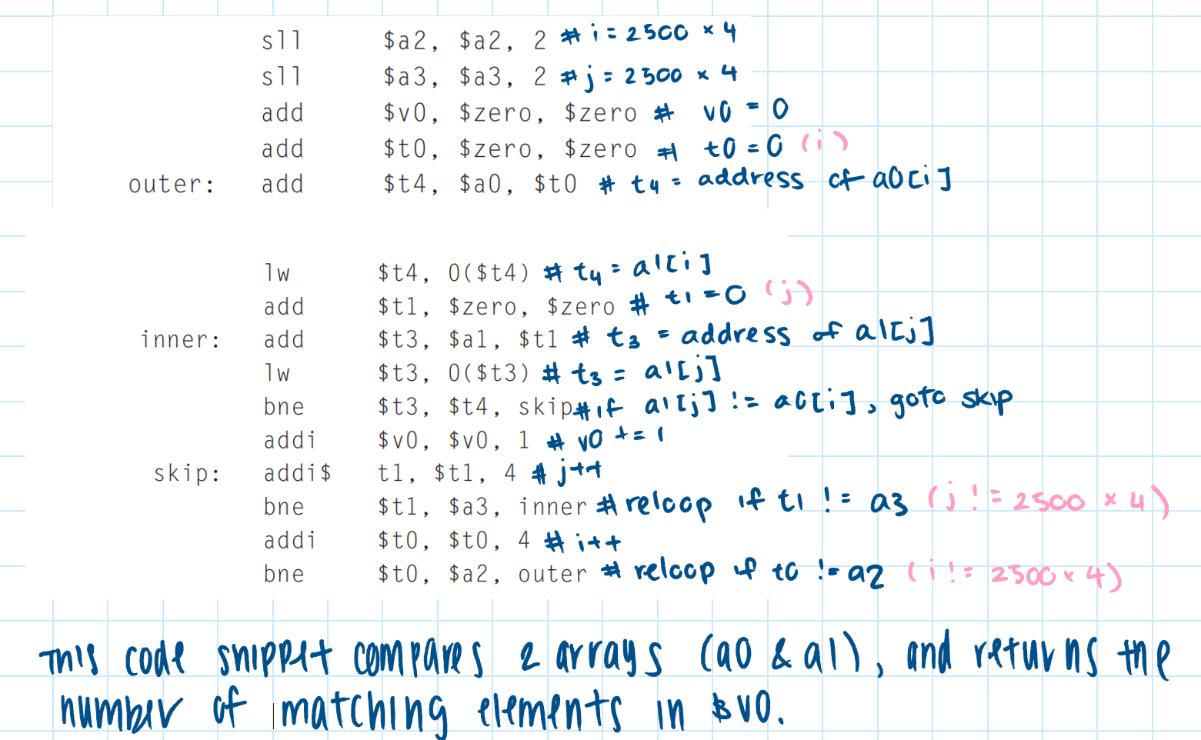
srl $t0, $t0, 15

2.15)



2.29)

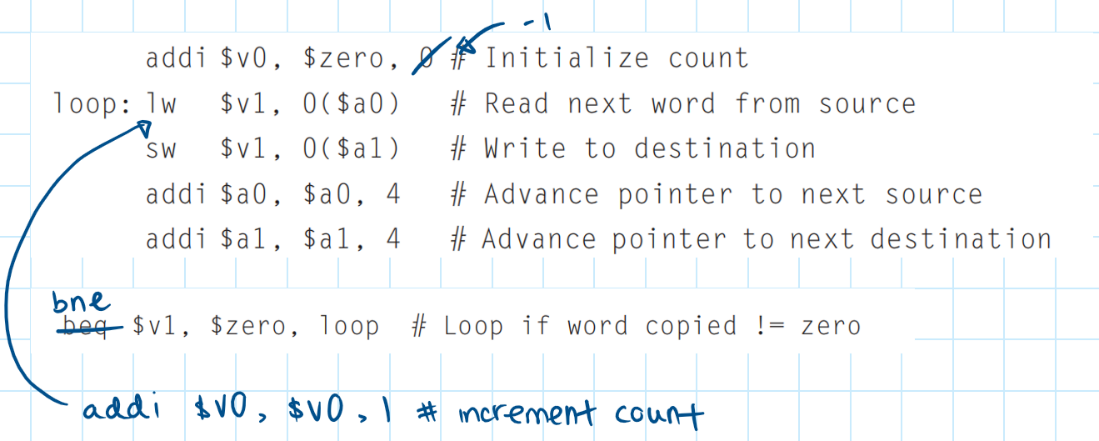
2.30)



2.32)

b = 25 | a -> ori $t1 $t0 25

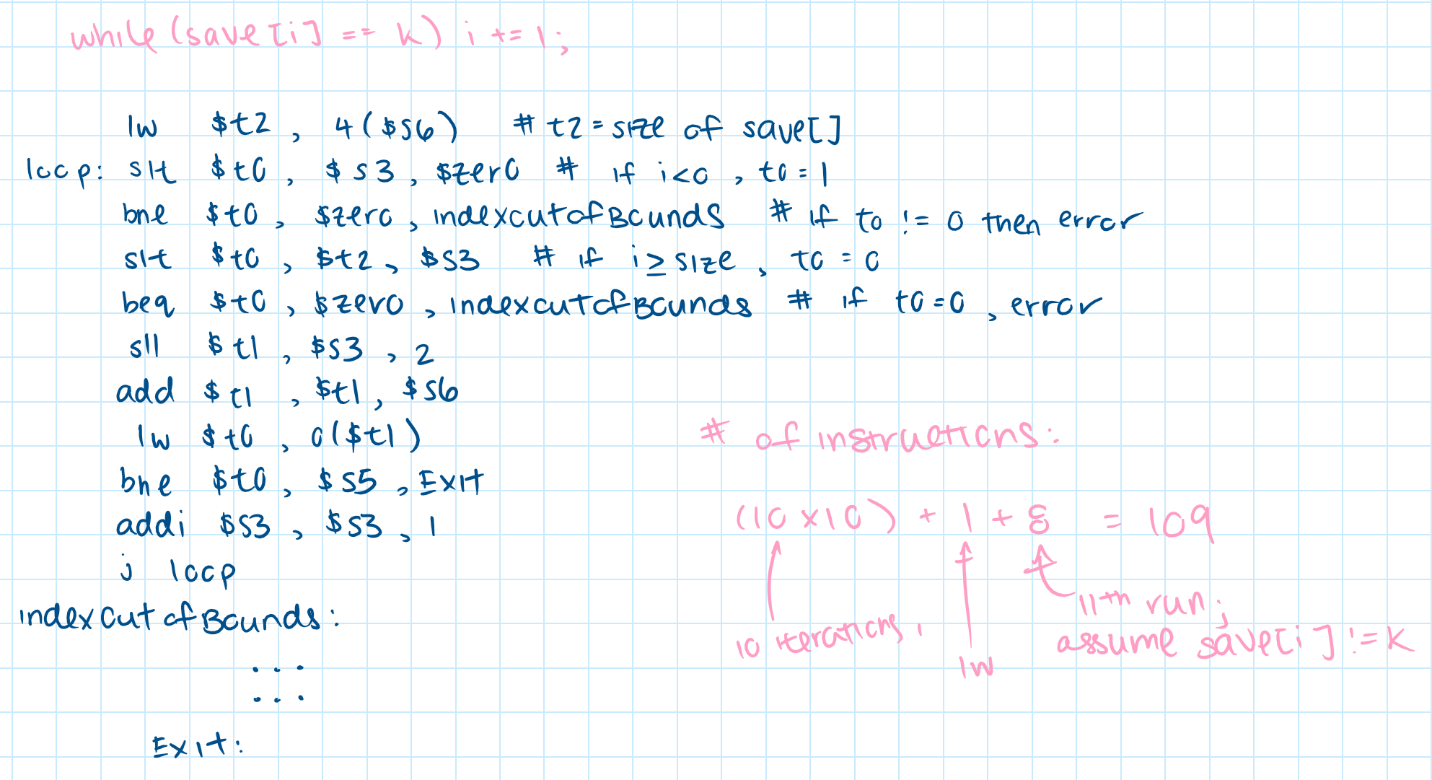
2.34)



2.37)

|  |  |  |
| --- | --- | --- |
| Pseudoinstructions | Purpose | MIPS Instructions |
| move $t1, t2 | $t1 = $t2 | add $t1, $t2, $zero |
| clear $t0 | $t0 = 0 | add $t0, $zero, $zero |
| beq $t1, small, L | if ($t1 = small) go to L | li $at, small  beq $t1, $at, L |
| beq $t2, big, L | if ($t2 = big) go to L | li $at, big  beq $t2, $at, L |
| li $t1, small | $t1 = small | addi $t1, $zero,small |
| li $t2, big | $t2 = big | addi $t1, $zero, big |
| ble $t3, $t5, L | if ($t3 <= $t5) go to L | slt $at, $t5, $t3  bne $at, $zero, L |
| bgt $t4, $t5, L | if ($t4 > $t5) go to L | slt $at, $t5, $t4  bne $at, $zero, L |
| bge $t5, $t3, L | if ($t5 >= $t3) go to L | slt $at, $t5, $t3  beq $at, $zero, L |
| addi $t0, $t2, big | $t0 = $t2 + big | li $at, big  add $t0, $t2, $at |
| lw $t5, big($t2) | $t5 = memory[$t2 + big] | li $at, big  add $at, $at, $t2  add $t5, $t2, $at |

2.46)



2.47)

