Exercise 4:

A

If "li" command has op code "001" then: `RR $r` is the same with `li <($r register address * 4) + 2>`

B

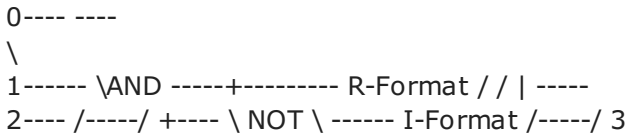
A single R-Format opcode "contains" (can be used) 4 commands due to the "funct" which is 2 bits long.

We can't have more than 4 r-format commands as we have no more available bits

So the commands will be:

op=111 funct=00
op=111 funct=01
op=111 funct=10
op=111 funct=11

C



D

R-Format Commands:

op=110 funct=00
op=110 funct=01
op=110 funct=10
op=110 funct=11

op=111 funct=00
op=111 funct=01
op=111 funct=10
op=111 funct=11The reason that there is no beqi or bnei is that the the commands would load the immediates to the registers

Exercise 5:

5.3

NOTES: i = $16, j = $17, $at = $1

i

beq $16, $17, L1

ii

bne $16, $17, L1

iii

slt $at, $17, $16 bne $at, $0, L1

iv

beq $17, $16, L1 slt $at, $17, $16 bne $at, $0, L1

v

slt $at, $16, $17 bne $at, $0, L1

vi

beq $16, $17, L1 slt $at, $16, $17 bne $at, $0, L1

vii

addi $at, $0, CONST beq $16, $at, L1

viii

addi $at, $0, CONST bne $16, $at, L1

ix

slti $at, $16, CONST bne $at, $0, L1

x

addi $at, $0, CONST beq $16, $at, L1 slti $at, $16, CONST bne $at, $0, L1

xi

slti $at, $16, CONST beq $at, $0, L1

xii

5.2

```
addi $at, $0, CONST beq $16, $at, L1 slti $at, $16, CONST beq $at, $0, L1 addi $s3, $0, -1
Loop: addi $s3, $s3, 1 sll $t1, $s3, 2 add $t1, $t1, $s6 lw $t0, 0($t1) bne $t0, $s5, Loop j Exit
Exit: ...
```

Commands executed with old code: 1 initilization 6*9 on each loop 4 on last loop = 59 commands

Commands executed with new code: 1 initilization 5*9 on each loop 6 on last loop = 52 commands