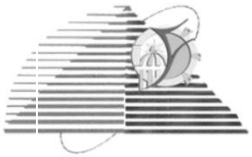
Fayoum University
Faculty of Engineering
Department of Electrical Engineering
Computer engineering systems CEE 216
Test Code



Second semester 2019/2020
Dr. Gihan Naguib
TA/ Muhammad Dakheel

- **Convert the following code into its binary equivalent, and use it to run your processor.**

```
set r1, 0x55          ; set reg[1] to 0x12102000
sset r1, 0x12
sset r1, 0x10
sset r1, 0x20
sset r1, 0x00
set r6, 0x22          ; set reg[6] to 0x22060211
sset r6, 0x06
sset r6, 0x02
sset r6, 0x11
eqv r3, r6, r1
sub r0,r0,r0    ; set reg[0] to 0, use as base
lw  r1, 0(r0)   ; reg[1] <- mem[0]
lw  r2, 1(r0)   ; reg[2] <- mem[1]
lw  r3, 2(r0)   ; reg[3] <- mem[2]
addi r4, r4, 10
sub r4,r4,r4    ; reg[4] <- 0, running total
add r4,r2,r4    ; reg[4]+ = A
slt r5,r2,r3    ; reg[5] <- A < B
beq r5,r0,2     ; if reg[5] = FALSE, go forward 2 instructions
add r2,r1,r2    ; A++
beq r0,r0,-5    ; go back 5 instructions
sw  r4, 0(r0)   ; mem[0] <- reg[4]
jal func
ror r6, r6, 8
lw  r4, 1(r0)
```

```
lw  r5, 2(r0)
add r6, r6, r5



beq r0,r0,-1    ; program is over, keep looping back to here

func: sub r0,r0,r0     ; set reg[0] to 0
lw  r1, 0(r0)     ; reg[1] <- mem[0]
lw  r2, 0(r1)     ; reg[2] <- mem[5]
lw  r3, 1(r1)     ; reg[3] <- mem[6]
and r4,r2,r3     ; reg[4] <- reg[2] AND reg[3]
or  r5, r2, r3     ; reg[5] <- reg[2] OR reg[3]
sw  r4, 1(r0)     ; mem[1] <- reg[4]
sw  r5, 2(r0)     ; mem[2] <- reg[5]
jr r7
```

```
Memory starting from address 0 contains:

00000001
00000001
0000000a
00000000
00000000
430a1f9b
728cd2e3
```

List output of all registers and locations of memory that have changed.