Zane Saul

1)

ble Rs, Rt, Label  
✓Branch on less than or equal;   
If (𝑅𝑠 ≤ 𝑅𝑡) then branch to label

SLT $at, $Rs, $Rt

BEQ $at, $0, Label

2)

abs Rd, Rs ✓Absolute;

Rd = absolute-value (Rs)

addu Rd, $0, Rs

bgez Rs, 1

sub Rd, $0, Rs

3)

slei Rd, Rs, Constant

✓Set less than or equal immediate;

If (𝑅𝑠 ≤ 𝐶𝑜𝑛𝑠𝑡𝑎𝑛𝑡) then 𝑅𝑑 = 1, else 𝑅𝑑 = 0.

bne Rt, Rs, yes

ori Rd, $0, 1

bet $0, $0, skip

yes: slt Rd, Rs, Rt  
skip:

4)

bgeo Rs, Label

✓Branch on greater than or equal to one; If (𝑅𝑠 ≥ 1) then branch to label

slt $at, Rs, 1

beq $at, $0, Label

5)

swapr Rs, Rt

✓Swaps the contents of registers Rs and Rt Rs🡨Rt || Rt🡨Rs

Xor $rs,$rs,$rt

Xor $rt,$rs,$rt

Xor $rs,$rs,$rt

6)

swapm (Rs), (Rt)

✓Swaps the contents of the memory locations pointed to by registers Rs and Rt M[Rs]🡨M[Rt] || M[Rt]🡨M[Rs]

sll $t0, $a1, 2

add $t1, $a0, $t0

lw $t3, 0($t1)

lw $14, 0($t1)

sw $14, 0($t1)

sw $13, 4($t1)