SAL instruction	Operation
`move x, y`	x ← (y)
`add x, y, z`	x ← (y) + (z)
`sub x, y, z`	x ← (y) − (z)
`mul x, y, z`	x ← (y) * (z)
`div x, y, z`	x ← (y)/(z)
`rem x, y, z`	x ← (y) mod (z)
`cvt x, y`	x ← (y), with type conversion
`not x, y`	x ← **NOT** (y), bitwise operation
`and x, y, z`	x ← (y) **AND** (z), bitwise operation
`or x, y, z`	x ← (y) **OR** (z), bitwise operation
`nand x, y, z`	x ← (y) **NAND** (z), bitwise operation
`nor x, y, z`	x ← (y) **NOR** (z), bitwise operation
`xor x, y, z`	x ← (y) **XOR** (z), bitwise operation
`xnor x, y, z`	x ← (y) **XNOR** (z), bitwise operation
`sll x, y, AMT`	x ← (y), logically left shifted by AMT bits
`srl x, y, AMT`	x ← (y), logically right shifted by AMT bits
`sra x, y, AMT`	x ← (y), arithmetically right shifted by AMT bits
`rol x, y, AMT`	x ← (y), rotated left by AMT bits
`ror x, y, AMT`	x ← (y), rotated right by AMT bits
`b label`	PC ← label
`j label`	PC ← label
`beq y, z, label`	if (y) = (z) then PC ← label
`bne y, z, label`	if (y) ≠ (z) then PC ← label
`blt y, z, label`	if (y) < (z) then PC ← label
`bgt y, z, label`	if (y) > (z) then PC ← label
`ble y, z, label`	if (y) ≤ (z) then PC ← label
`bge y, z, label`	if (y) ≥ (z) then PC ← label
`beqz y, label`	if (y) = 0 then PC ← label
`bnez y, label`	if (y) ≠ 0 then PC ← label
`bltz y, label`	if (y) < 0 then PC ← label
`bgtz y, label`	if (y) > 0 then PC ← label
`blez y, label`	if (y) ≤ 0 then PC ← label
`bgez y, label`	if (y) ≥ 0 then PC ← label
`la x, label`	x ← label