

# Resumen de Operaciones MIPS32

Adrian Cordero Diaz C.I.: 30.585.449

9 de Diciembre, 2025

## Operaciones de carga y almacenamiento

- **lw \$t1, 0(\$s0)** Cargar palabra (4 bytes) desde memoria.
- **sw \$t1, 0(\$sp)** Guardar palabra en memoria.
- **lb / lbu** Cargar byte (con o sin signo).
- **la \$a0, etiqueta** Cargar dirección de una etiqueta.

## Operaciones aritméticas

- **add \$t0, \$a1, \$t0** Suma.
- **sub \$t0, \$a2, \$a1** Resta.
- **mul \$v0, \$a0, \$v0** Multiplicación.
- **sll \$s0, \$t0, 2** Shift left lógico (multiplica por 4).
- **sra \$t0, \$t0, 1** Shift right aritmético (divide entre 2).
- **addi \$sp, \$sp, -8** Suma inmediata (ajuste de stack).

## Saltos y control de flujo

- **beq \$t1, \$a3, etiqueta** Branch if equal.
- **bgt \$t1, \$a3, etiqueta** Branch if greater than.
- **ble \$a1, \$a2, etiqueta** Branch if less or equal.

- **j etiqueta** Salto incondicional.
- **jal subrutina** Llamada a subrutina (guarda \$ra).
- **jr \$ra** Retorno de subrutina.

## Syscalls

- **li \$v0, 1** Imprimir entero (\$a0 contiene el valor).
- **li \$v0, 11** Imprimir carácter (\$a0 contiene el código ASCII).
- **li \$v0, 10** Salir del programa.

## Ejemplo: Factorial Recursivo

```

1 factorial:
2     addi $sp, $sp, -8
3     sw $a0, 0($sp)
4     sw $ra, 4($sp)
5
6     beq $a0, $zero, base_case
7
8     addi $a0, $a0, -1
9     jal factorial
10
11    lw $a0, 0($sp)
12    mul $v0, $a0, $v0
13    j end_factorial
14
15 base_case:
16     li $v0, 1
17
18 end_factorial:
19     lw $ra, 4($sp)
20     addi $sp, $sp, 8
21     jr $ra

```