

## – CATE ORGANIZACIÓN DE COMPUTADORAS 2025 –

Alumna: ANDACABA, Madai

Profesor: TRINI, Santiago

<i><b>Lenguaje Assembler</b></i>	<i><b>Lenguaje de Máquina</b></i>	<i><b>Pseudocódigo</b></i>
syscall (Nada)	0000	<b><u>No hace nada</u></b>
lw	0001	Lee el valor de un espacio en la memoria
sw	0010	Guarda el valor de un registro en un espacio de la memoria
addi	0011	Suma un valor con un número fijo (inmediato)
and	0100	Funciona como una compuerta AND bitwise
or	0101	Funciona como una compuerta OR bitwise
nand	0110	Funciona como una compuerta NAND bitwise
xor	0111	Funciona como una compuerta XOR bitwise

add	1000	Suma el primer registro con el segundo, y guarda el resultado en el tercer registro
sub	1001	Resta el primer registro con el segundo, y guarda el resultado en el tercer registro
beq	1010	Suma o resta el PC, con la condición de que 2 registros sean iguales, esto mediante la resta y la señal de Zero
	1011	NADA
	1100	NADA
	1101	NADA
	1110	NADA
Reiniciar	1111	OPCODE ficticio, activa la entrada de Restart de los registros A, B, C y D

<i>Registro</i>	<i>Dirección</i>	<i>¿Qué es?</i>
A	00	Primer registro multiusos
B	01	Segundo registro multiusos
C	10	Tercer registro multiusos

D	11	Cuarto registro multiusos
PC	—	Registro del Program Counter, guarda el valor de este y se incrementa en 1 por defecto
IR	—	Guarda las instrucciones en lenguaje máquina
Unidad de Control	—	Contiene los valores de control, se ubican mediante los opcodes

## **FIBONACCI**

<b><i>Instrucción</i></b>	<b><i>Binario</i></b>	<b><i>Hexadecimal</i></b>	<b><i>Pseudocódigo</i></b>
main:			
addi a, a, 0	0011 00 00 00000000	3000	Se asigna el valor 0 a A
addi b, b, 1	0011 01 01 00000001	3501	Se asigna el valor 1 a B
addi c, c, 0	0011 10 10 00000000	3A00	Se asigna el valor 0 a C
addi d, d, 233	0011 11 11 11101001	3FE9	Se asigna el valor 233 a D
loop:			
beq c, d, reiniciar	1010 10 11 00000100	AB04	Si C es igual a D (233), salta a reiniciar
add c, a, b	1000 00 01 10 000000	8180	$c = a + b$
addi a, b, 0	0011 01 00 00000000	3400	$a = b // a = b + 0$
addi b, c, 0	0011 10 01 00000000	3900	$b = c // b = c + 0$
beq a, a, loop	1010 00 00 11111011	A0FB	Salta a la etiqueta loop
reiniciar:			
reiniciar	1111 0000 0000 0000	F000	Reinicia

## TEST DE MEMORIA

<i><b>Instrucción</b></i>	<i><b>Binario</b></i>	<i><b>Hexadecimal</b></i>	<i><b>Pseudocódigo</b></i>
main:			
addi a, a, 32	0011 00 00 00100000	3020	A = 32
addi b, b, 68	0011 01 01 01000100	3544	B = 68
add d, a, b	1000 00 01 11 000000	81C0	D = A + B
sw d, 0(c)	0010 10 11 00000000	2B00	Guarda en el espacio 0 (c) el valor de D
sub d, d, d	1001 11 11 11 000000	9FC0	D = D - D (Establece en 0)
lw c, 0(d)	0001 11 10 00000000	1E00	Carga en c el valor del espacio 0 (d)
NOP	0000 0000 0000 0000	0000	Delay para mostrar más tiempo en pantalla
sw d, 0(d)	0010 11 11 00000000	2F00	Carga 0 en el valor del espacio 0
restart	1111000000000000	F000	Reiniciar