



# Proyecto Unidad 3 y 4

## Arquitectura de Computadores

### Compilador SimpleRISC

Ricardo Pérez  
riperez@utalca.cl

## Problema a resolver

Debido a la falta de disponibilidad de un compilador eficiente para el módulo Arquitectura de Computadores, su profesor les pide a sus mejores estudiantes del 2022, que diseñen un compilador Simple RISC funcional, para utilizarlo en los próximos años.

Para ello es necesario:

1. Implementar un compilador SimpleRisc completamente funcional. Es decir, todas las instrucciones del set de instrucciones deben ser funcionales. Pueden consultar la tabla a continuación.

Inst.	Código	Formato	Inst.	Código	Formato
add	00000	add rd, rs1, (rs2/imm)	lsl	01010	lsl rd, rs1, (rs2/imm)
sub	00001	sub rd, rs1, (rs2/imm)	lsr	01011	lsr rd, rs1, (rs2/imm)
mul	00010	mul rd, rs1, (rs2/imm)	asr	01100	asr rd, rs1, (rs2/imm)
div	00011	div rd, rs1, (rs2/imm)	nop	01101	nop
mod	00100	mod rd, rs1, (rs2/imm)	ld	01110	ld rd, imm[rs1]
cmp	00101	Cmp rs1, (rs2/imm)	st	01111	st rd, imm[rs1]
and	00110	and rd, rs1, (rs2/imm)	beq	10000	beq offset
or	00111	or rd, rs1, (rs2/imm)	bgt	10001	bgt offset
not	01000	not rd, (rs2/imm)	b	10010	b offset
mov	01001	mov rd, (rs2/imm)	call	10011	call offset
			ret	10100	ret

2. Implemente el algoritmo necesario para detectar conflicto entre dos instrucciones. (Pueden consultar el ejemplo de Algoritmo mostrado en la presentación 15 durante clases).
3. Otorgue la posibilidad al compilador de reordenar el código, de manera que se puedan evitar peligros de datos (Data-Hazards).
4. Implemente las llamadas ranuras de retardo (delay-slots), de manera que al ejecutar una instrucción de salto, se ejecuten 2 instrucciones dentro de la pipeline que no afecten el resultado de la instrucción después del salto.

Pueden utilizar el lenguaje de programación de su preferencia y la idea es presentar el resultado de las instrucciones SimpleRisc ingresadas por el usuario en pantalla.

La resolución del proyecto será en dúos y no es necesario una interfaz gráfica. Puede mostrar los resultados por consola.

## Referencias:

- Presentaciones de la Unidad disponibles en Educandus.
- Modern Compiler Implementation in Java (Libro disponible en Educandus).
- Modern Compiler Implementation in C (Libro disponible en Educandus).