*Abdelhafid Boussouf University, Mila*
*Institute of Mathematics and computer science*
*Department of Computer Science*
*2nd Year Informatics*

*Year: 2025/2026*
*Module: Computer Architecture*

# Directed Work N° 03

**Exercise1:**

1- Indicate data hazards and their type for each of the following two sequences of instructions.

a. I1 : lw $t1, 40($t0)
I2 : add $t0, $t2,$t2
I3 : sw $t0, 50($t1)

b. I1 : lw $t1, -8($t1)
I2 : sw $t1, -8($t1)
I3 : add $t1, $t1,$t1

2- Assume there is no forwarding in this pipelined processor. Add nop instructions to eliminate hazards

3- Assume there is full forwarding. Add nop instructions to eliminate hazards

**Exercise2:**

Find the hazards in this code segment and reorder the instructions to avoid any pipeline stalls.

```
lw $t1, 0($t0)
lw $t2, 4($t0)
add $t3, $t1,$t2
sw $t3, 12($t0)
lw $t4, 8($t0)
add $t5, $t1,$t4
sw $t5, 16($t0)
```

**Exercise3:**

Identify all the RAW data dependencies in the following code. Which dependencies are data hazards that will be resolved by forwarding? Which dependencies are data hazards that will cause a stall? Show the forwarding paths and stalled cycles if any.

```
add $3, $4, $2
sub $5, $3, $1
lw $6, 200($3)
add $7, $3, $6
```

**Exercise4:**

For this problem, assume that all branches are perfectly predicted (this eliminates all control hazards). If we only have one memory (for both instructions and data), there is a structural hazard every time. What is the total execution time of this instruction sequence in the five-stage pipeline? We have seen that data hazards can be eliminated by adding nops to the code. Can you do the same with this structural hazard? Why?

a.
```
    lw $1,40($6)
    beq $2, $0, Label; assume $2=$0
    sw $2, 50($2)
Label: add $2, $3, $4
    sw $3, 50($4)
```

b.
```
lw $5,-16($5)
sw $4, -16($4)
lw $3,-20($4)
beq $2, $0, Label;assume $2!=$0
add $5, $1, $4
```