



# Université d'Alger 1

Faculté des Sciences

Département Mathématiques et Informatique

## Rattrapage L2-ADO

11-Septembre-2021

Nom : Prénom : Groupe : Matricule :		/ 20
--	--	------

### Exercice 1 : (15 points)

Pour chacune des pseudo-instructions suivantes, donnez une **séquence minimale** d'instructions MIPS réelles pour accomplir la même chose. Vous ne pouvez utiliser que le registre \$at en tant que registre temporaire pour vos transformations

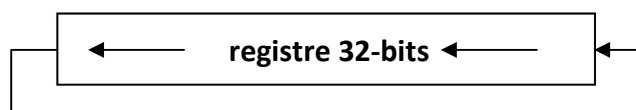
(3 points) a) `abs $s1, $s2` # \$s1 = valeur absolue de \$s2

(3 points) b) `addiu $s1, $s2, imm32` # imm32 est un immediat 32-bits

(3 points) c) `bleu $s1, $s2, Label` # saut si inférieur ou égal (non signé)

(3 points) d) `bge $s1, imm32, Label` # imm32 est un immediat 32-bits

(3 points) e) `rol $s1, $s2, 5` # rol = rotation à gauche de \$s2 par 5 bits



## Exercice 2 : (5 points)

Le code suivant traite deux tableaux et produit un résultat important dans le registre \$v0. Supposons que chaque tableau se compose de N mots, les adresses de base des tableaux A et B sont stockées dans les registres \$a0 et \$a1 respectivement, et leurs tailles sont stockées dans les registres \$a2 et \$a3, respectivement.

```
        sll    $a2, $a2, 2
        sll    $a3, $a3, 2
        addu   $v0, $zero, $zero
        addu   $t0, $zero, $zero
outer:   addu   $t4, $a0, $t0
        lw     $t4, 0($t4)
        addu   $t1, $zero, $zero
inner:   addu   $t3, $a1, $t1
        lw     $t3, 0($t3)
        bne    $t3, $t4, skip
        addiu   $v0, $v0, 1
skip:    addiu   $t1, $t1, 4
        bne    $t1, $a3, inner
        addiu   $t0, $t0, 4
        bne    $t0, $a2, outer
```

1/ Décrivez ce que fait le code ci-dessus et ce qui sera retourné dans le registre \$v0.

(2 points)

2/ Donnez une version en langage C du code ci-dessus

(3 points)