

Correction TD 7

Solution exercice 1:

```
.data
tabelau :.space 20
espace :.asciiz " "
.text
addi $s1, $zero, 4
addi $s2, $zero, 10
addi $s3, $zero, 12
addi $s4, $zero, 25
addi $s5, $zero, 64

addi $t0, $zero, 0
sw $s1, tabelau($t0)

addi $t0, $t0, 4
sw $s2, tabelau($t0)

addi $t0, $t0, 4
sw $s3, tabelau($t0)

addi $t0, $t0, 4
sw $s4, tabelau($t0)

addi $t0, $t0, 4
sw $s5, tabelau($t0)

addi $t0, $zero, 0

boucle :
beq $t0, 20, fin
lw $t6, tabelau($t0)
addi $t0, $t0, 4

li $v0, 1
move $a0, $t6
syscall

li $v0, 4
la $a0, espace
syscall
b boucle

fin:
li $v0, 10
syscall
```

Solution exercice 2:

```
.data
tab:.space 40
```

entree:.asciiz "donnez la dimension de votre tableau SVP:"

entree2:.asciiz "Inserer l'element SVP:"

espace:.asciiz " "

.text

li \$v0, 4

la \$a0, entree

syscall

li \$v0, 5

syscall

move \$t2, \$v0

addi \$t1, \$zero, 1

la \$t0, tab

compteur :

bgt \$t1, \$t2, afficher

li \$v0, 4

la \$a0, entree2

syscall

li \$v0, 5

syscall

move \$t3, \$v0

sw \$t3, (\$t0)

addi \$t0, \$t0, 4

addi \$t1, \$t1, 1

b compteur

afficher:

li \$t1, 1

la \$t0, tab

boucle :

bgt \$t1, \$t2, fin

lw \$t4, (\$t0)

addi \$t0, \$t0, 4

addi \$t1, \$t1, 1

li \$v0, 1

move \$a0, \$t4

syscall

li \$v0, 4

la \$a0, espace

syscall

b boucle

fin:

li \$v0, 10

syscall

Solution exercice 3:

```

.data
tab:.space 40
entree:.asciiz "donnez la dimension de votre tableau SVP:"
entree2:.asciiz "Insérer l'élément SVP:"
sortie1:.asciiz "le max du tableau est: "
saut:.asciiz "\n"
sortie2:.asciiz "le min du tableau est: "
.text
#affiche le premier message
li $v0, 4
la $a0, entree
syscall
#lire la première valeur
li $v0, 5
syscall
move $t2, $v0
#initialiser t1 et t0
addi $t1, $zero, 1
la $t0, tab

li $v0, 4
la $a0, entree2
syscall
#lire la valeur tapée au clavier et la stocke dans t3
li $v0, 5
syscall
move $t3, $v0
sw $t3, ($t0)
#initialiser le min et le max
move $t4, $t3
move $t5, $t3
addi $t1, $t1, 1
addi $t0, $t0, 4

compteur :
bgt $t1, $t2, fin
# affiche le deuxième message
li $v0, 4
la $a0, entree2
syscall
#lire la valeur tapée au clavier et la stocke dans t3
li $v0, 5
syscall
move $t3, $v0
sw $t3, ($t0)
#poser les valeurs des booléens t6 et t7
sgt $t6,$t3, $t4
slt $t7, $t3, $t5
#tester les valeurs de t6 et de t7 avec zero et poser les nouvelles valeurs de min et max
beqz $t6, suite

```

```
move $t4, $t3
suite:
beqz $t7, continuer
move $t5, $t3
```

```
continuer:
#incrémenter le compteur
addi $t1, $t1, 1
#incrémenter l'indice du tableau
addi $t0, $t0, 4
b compteur
```

```
fin:
li $v0, 4
la $a0, sortie1
syscall
li $v0, 1
move $a0, $t4
syscall
li $v0, 4
la $a0, saut
syscall
li $v0, 4
la $a0, sortie2
syscall
li $v0, 1
move $a0, $t5
syscall
```

```
li $v0, 10
syscall
```

Solution exercice 4 :

```
.data
tab:.space 40
entree:.asciiz "donnez la dimension de votre tableau SVP:"
entree2:.asciiz "Inserer l'element SVP:"
sortie1:.asciiz "le max du tableau est: "
saut:.asciiz "\n"
sortie2:.asciiz "le min du tableau est: "
sortie3:.asciiz "la position du max est: "
sortie4:.asciiz "la position du min est: "
.text
#affiche le premier message
li $v0, 4
la $a0, entree
syscall
#lire la première valeur
li $v0, 5
syscall
```

```
move $t2, $v0
#initialiser t1 et t0
addi $t1, $zero, 1
la $t0, tab
```

```
li $v0, 4
la $a0, entree2
syscall
#lire la valeur tapée au clavier et la stocke dans t3
li $v0, 5
syscall
move $t3, $v0
sw $t3, ($t0)
#initialiser le min et le max
move $t4, $t3
move $t5, $t3
move $s1, $t1
move $s2, $t1
addi $t1, $t1, 1
addi $t0, $t0, 4
```

```
compteur :
bgt $t1, $t2, fin
# affiche le deuxième message
li $v0, 4
la $a0, entree2
syscall
#lire la valeur tapée au clavier et la stocke dans t3
li $v0, 5
syscall
move $t3, $v0
sw $t3, ($t0)
#poser les valeurs des booléens t6 et t7
sgt $t6,$t3, $t4
slt $t7, $t3, $t5
#tester les valeurs de t6 et de t7 avec zero et poser les nouvelles valeurs de min et max
beqz $t6, suite
move $t4, $t3
move $s1, $t1
suite:
beqz $t7, continuer
move $t5, $t3
move $s2, $t1
continuer:
#incrémenter le compteur
addi $t1, $t1, 1
#incrémenter l'indice du tableau
addi $t0, $t0, 4
b compteur
```

```
fin:
li $v0, 4
la $a0, sortie1
syscall
li $v0, 1
move $a0, $t4
syscall
li $v0, 4
la $a0, saut
syscall
li $v0, 4
la $a0, sortie2
syscall
li $v0, 1
move $a0, $t5
syscall
li $v0, 4
la $a0, saut
syscall
li $v0, 4
la $a0, sortie3
syscall
li $v0, 1
move $a0, $s1
syscall
li $v0, 4
la $a0, saut
syscall
li $v0, 4
la $a0, sortie4
syscall
li $v0, 1
move $a0, $s2
syscall

li $v0, 10
syscall
```