

**TD 4 : Architecture des ordinateurs**  
**Solutionnaire**

**Exercice 1**

Donner les numéros et le code binaire des registres généraux suivants :

Registres	Numéro	Code binaire sur 5 bits
\$t0,\$t1,\$t2,...,\$t7,\$t8,\$t9		
\$s0,...,\$s7		
\$a0,...,\$a3		
\$v0,...,\$v1		
\$zero, \$gp, \$sp		
\$ra		

**Exercice 2**

Commenter le code suivant :

```
add $t0,$t1,$t2
sub $t2,$t3,$t4
addi $t2,$t3, 5
addu $t1,$t6,$t7
subu $t1,$t6,$t7
mult $t3,$t4
div $t5,$t6
mfhi $t0
mflo $t1
move $t2,$t3
```

**Exercice 3**

- Etant donnée la séquence suivante :

$$f = (g + h) - (i + j);$$

Avec l'affectation respective des variables f, g, h, i, et j aux registres \$s0, \$s1, \$s2, \$s3, et \$s4. Quel est le code d'assemblage MIPS ?

- Quelle est la différence entre les instructions de saut suivantes ?

```
j etiquette
jr $31
jal proc1
```

- Supposons que A est un ensemble de 100 mots et que le compilateur a associé les variables g et h aux registres \$s1 et \$s2. Supposons aussi que l'adresse du premier élément du tableau A est en \$s3.

$$g = h + A[8];$$

Ecrire le code d'assemblage MIPS correspondant.

**Exercice 4**

- Ecrire l'instruction MIPS correspondante à chaque. Dans tous les cas l'étiquette "suite" figurera dans l'instruction de branchement.

**TD 4 : Architecture des ordinateurs**  
**Solutionnaire**

1. if \$7 < 0 then  
    instructions  
end\_if  
suite:

2. if \$15 ≥ 0 then  
    instructions  
end\_if  
suite:

3. if \$11 ≤ 0 then  
    instructions  
end\_if  
suite:

4. if \$2 == \$10 then  
    instructions  
end\_if  
suite:

- Ecrire les deux instructions MIPS de chaque condition de test ci-dessous. On utilisera le registre \$1 comme registre intermédiaire :

5.

6. if \$21 < \$17 then  
    instructions  
end\_if  
suite:

7. if \$18 ≥ \$13 then  
    instructions  
end\_if  
suite:

8. if \$21 > \$17 then  
    instructions  
end\_if  
suite:

9. if \$18 ≤ \$13 then  
    instructions  
end\_if  
suite:

**Exercice 5**

Ecrire en langage d'assemblage MIPS le fragment C suivant :

```
if (t1 < t2)
{
    t3 = t1 ;
}
else
```

**TD 4 : Architecture des ordinateurs**  
**Solutionnaire**

```
t3 = t2 ;
}
```

Même question pour les fragments suivants :

```
t2 = 0;
while (t1 > 0)
{
    t2 = t2 + t1;
    t1 = t1-1 ;
}
```

```
for (t1=1 ; t1< N ; t1++)
{
    t2 = t2 +2×t1;
}
```

**Exercice 6**

Ecrire en langage d'assemblage le fragment de code dans les cas suivants :

- Affichage d'un entier.
- Affichage d'une chaîne de caractères.
- Lecture d'un entier.
- Lecture d'une chaîne de caractère.
- Mettre fin à l'exécution du programme

**Exercice 7**

- Quelle est l'instruction MIPS du code machine suivant : 0x00AF8020
- Compléter le tableau suivant en précisant le format R, I ou J des opcodes puis le langage machine des instructions à droite :

opcode	format	Langage machine						Commentaire
add								add \$s1,\$s2,\$s3
sub								sub \$s1,\$s2,\$s3
lw								lw \$s1, 100(\$s2)
sw								sw \$s1, 100(\$s2)
and								and \$s1,\$s2,\$s3
or								or \$s1,\$s2,\$s3
beq								beq \$s1,\$s2, 100
bne								bne \$s1,\$s2, 100
slt								slt \$s1,\$s2, \$s3
j								j 10000
Taille du champ								
R								
I								
J								