



UNIVERSITE M'Hamed BOUGARA – BOUMERDES
FACULTE DES SCIENCES
DEPARTEMENT D'INFORMATIQUE

Module : Architecture des Ordinateurs

Responsable : Mokrani Hocine

Filière : Licence Informatique

Semestre : 3

Série N°3 : Assembleur MIPS 32 bits

Exercice 1 *

Quel est le code assembleur MIPS pour le code C ci-dessous ?

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

Supposons que les variables **f**, **g**, **h**, **i** et **j** peuvent être assignées aux registres **\$16** à **\$20** respectivement. En pratique, c'est au compilateur que revient cette tâche délicate d'assignement des variables).

Exercice 2 *

Quel est le code assembleur MIPS pour le code C ci-dessous ?

if (i==j) f=g+h; else f=g-h;

Supposons que **f**, **g**, **h**, **i** et **j** correspondent aux registres **\$16** à **\$20** respectivement.

Exercice 3 *

Quel est le code assembleur MIPS pour le code C ci-dessous ?

T[i]=h+T[i] ;

Supposons que:

1. **T** est un tableau d'entiers.
2. La variable **h** est dans le registre **\$18**.
3. La variable **i** est dans le registre **\$19**, et que le tableau débute à l'adresse **Tstart**.

Exercice 4 **

Traduire le code C suivante en langage assembleur MIPS.

While (stock[i]==k) {i=i+j;}

Supposons que **i**, **j** et **k** correspondent aux registres **\$19** à **\$21**, le tableau **stock** débute à **Sstart** et le registre **\$10** contient la valeur **4**.

Exercice 5**

Traduire le code C suivante en langage assembleur MIPS.

For (i=0; i<1000; i++) {C[i] = A[i] + B[i];}

Supposons que les adresses des variables soient :

24000–27996	vecteur A
28000–31996	vecteur B
32000–35996	vecteur C
36000	constante 0
36004	constante 3996

Exercice 6 (D-à-D)

Donner le code pour la procédure suivante

```
change(int v[], int k){  
    int temp;  
    temp = v[k];  
    v[k]=v[k+1];  
    v[k+1]=temp;  
}
```

Les paramètres **v** et **k** sont alloués à **\$4** et **\$5**, une variable **temp** en **\$15** et **\$2** contient la base du tableau

NB : les adresses de mots contigus différent de 4 et non de 1.

Rappels

Pour traduire du C en assembleur :

- Allouer des registres aux variables du programme.
- Produire du code pour le corps de la procédure.
- Préserver les registres à travers l'appel de la procédure

Convention MIPS : pour le passage de paramètres sont utilisés \$4 à \$7