

TD N°2

(Programmation en langage d'assemblage du CPU 8086)

(Traduction des structures algorithmiques en langage d'assemblage.)

Exercice 1 :

Ecrire les programmes en langage d'assemblage nécessaires pour exécuter les structures algorithmiques classiques suivantes :

Si ... Alors ... Sinon ...

```
Si (ax=1)
    bx ← 10;
Sinon
    debut
        bx ← 0;
        cx ← 10;
    Fin
EndIf;
```

La boucle Pour ...

```
bx ← 0 ;
Pour K = 0 jusqu'à 10
    bx ← bx + K ;
```

La boucle tant que ...

```
bl = 5 ;
Tant que bl > 0 faire
    Debut
        bl = bl-1
        ax ← al*bl
    Fin
```

La boucle Répéter

```
ax ← FF ;
bx ← 0 ;
Répéter
    debut
        ax ← ax -14 ;
        bx ← bx+ax ;
    fin
jusqu'à ax <= 0
```

Le test à plusieurs alternatives...

```
Switch(BX){
    case 1: ax ← 1 ; break;
    case 2: ax ← 5 ; break ;
    case 3: ax ← 10; break;
    default: ax ← 0;
}
```

Exercice 2 :

Ecrire les programmes en langage d'assemblage correspondant aux programmes Pascal suivants.
On supposera que les variables a, b et c sont initialement dans les registres AX, BX et CX.

a) if a+b > c then
 begin
 a := a+b ;
 c:= c+1;
 end;
 else
 begin
 a := a-b ;
 c:= c-1 ;
 end;
 end;

b) c:=0 ;
 for a:= 1 to 10 do
 c:= c+a ;
 end;
c) while a < 30 do
 begin
 a:= a+b ;
 c:= c+1
 end;
 end;

Devoir :

Ecrire les programmes en langage d'assemblage correspondant à l'algorithme d'Euclide pour le calcul du PGDC ci-contre :

```
b := 21;
c := 35;
while b > c do
begin
    if b > c then
        b := b-c;
    else
        c := c-b;
end;
```