

# UNIVERSITE CATHOLIQUE DE L'AFRIQUE DE L'OUEST

## UNITE UNIVERSITAIRE AU TOGO

Année : 2022 - 2023  
Professeur : M. SEWAVI  
Module : Algorithmes et structures de données

### Fiche de TD numéro 1

#### Exercice 1

1. Pour chacun des fonctions  $T_i(n)$  suivant, déterminer sa complexité asymptotique dans la notation Grand-O.
  - a.  $T_1(n) = 6n^3 + 10n^2 + 5n + 2$
  - b.  $T_2(n) = 3 \log_2 n + 4$
  - c.  $T_3(n) = 7k + 2$
  - d.  $T_4(n) = 2^n + 6n^2 + 7n$
  - e.  $T_5(n) = 2 \log_{10} k + kn$
  - f.  $3 \log_2 n + n$

#### Exercice 2

Soit les trois programmes suivants

Déterminer (en fonction de  $n$  à  $O()$  près) leur complexité en nombre d'opérations

Algorithme 1	Algorithme 2	Programme 1
Pour i allant de 1 à n faire Pour j allant de 1 à i faire Opération	Pour i allant de 10 à n-10 faire Pour j allant de i-10 à i+10 faire Opération	i = 0 j = 0 while(i < n) { if( i % 2 == 0) { j = j + 1 } else { j = j / 2 } i = i + 1 }

**Algorithme 3** $i \leftarrow 1; j \leftarrow 1$ **Tant que**  $j \leq n$  **faire****Si**  $i \leq m$ **Alors** $i \leftarrow i + 1$ **Sinon** $j \leftarrow j + 1$ **Fin Si****Fin Tant que****Algorithme 4** $i \leftarrow 1; j \leftarrow 1$ **Tant que**  $j \leq n$  **faire****Si**  $i \leq m$ **Alors** $i \leftarrow i + 1$ **Sinon** $j \leftarrow j + 1$  $i \leftarrow 1$ **Fin Si****Fin tant que****Exercice 3**

Considérer les deux algorithmes A1 et A2 avec leurs temps d'exécution  $T_1(n) = 9n^2$  et

$T_2(n) = 100n + 96$  respectives.

1. Déterminer la complexité asymptotique des deux algorithmes dans la notation Grand-O. Quel algorithme a la meilleure complexité asymptotique ?
2. Montrer que vos solutions sont correctes en spécifiant un  $c$  et un  $n_0$  par algorithme afin que la relation suivante soit satisfaite :  
 $O(f) = \{g | \exists c > 0 : \exists n_0 > 0 : \forall n \geq n_0 : g(n) \leq cf(n)\}$
3. Calculer les temps maximaux d'exécution des deux algorithmes  $T_i(n)$  pour  $n = 1, n = 3, n = 5, n = 10, n = 14$ .
4. Ébaucher les graphes des deux fonctions  $T_i$  dans un même système de coordonnées (abscisse  $n$ , ordonnée  $T_i(n)$ ).
5. Pour quelles longueur de donnée  $n$  quel algorithme est le plus efficace ?
6. Quelle est la complexité asymptotique de l'algorithme suivant ? Quelle règle avez-vous appliquée ?

début

appeler A1 {Ici l'algorithme 1 est exécuté. }

appeler A2 {Ici l'algorithme 2 est exécuté. }

fin

**Exercice 4**

1. Montrer que  $f(n) = 2n^2 - n + 1$  est  $O(n^2)$ .
2. On veut comparer les implémentations du tri par insertion et du tri par fusion sur la même machine. Pour un nombre  $n$  d'éléments à trier, le tri par insertion demande  $8n^2$  étapes alors que le tri par fusion en demande  $64n \lg n$ . Quelles sont les valeurs de  $n$  pour lesquelles le tri par insertion l'emporte sur le tri par fusion ?



### Exercice 5

Quel est le temps d'exécution (asymptotique) de chacun des algorithmes suivants, en fonction de  $n$  ? Justifiez vos réponses.

a.

```
for i = 1 to n do
  for j = 1 to  $2n + 1$  do
    print (« Hello World »)
  end for
end for
```

b.

```
for i = 1 to 10 do
  for j = 1 to n do
    print (« Hello World »)
  end for
end for
```

c.

```
for i = 1 to n do
  for j = i to n do
    print (« Hello World »)
  end for
end for
```

d.

```
for i = 1 to n do
  for j = 1 to  $2 * i + 1$  do
    print (« Hello World »)
  end for
end for
```

e.

```
for i = 1 to  $n * n$  do
  for j = 1 to i do
    print (« Hello World »)
  end for
end for
```

f.

```
for (i = 0 to m) do
  t ← 1
  while (t < m) do
    print (« Hello world »)
    t ←  $t * 2$ 
  end while
end for
```