

EA4 – Éléments d’algorithmique

TD n° 2

Exercice 1 : Ordres de grandeur

1. Montrer que $\frac{1}{2}n^2 - 3n \in \Theta(n^2)$.
2. Montrer que $5n^3 \notin \Theta(n^4)$.
3. Que peut-on dire plus généralement de deux polynômes P et Q ? ($P \in \Theta(Q)$ si et seulement si ...)
4. Est-ce que $2^{n+1} \in \Theta(2^n)$? Et 2^{2n} ?
5. Comparer $n!$ à 2^n et à n^n .

Soient $f, g : \mathbb{N} \rightarrow \mathbb{N}$.

6. Montrer que $f \in \Theta(g) \iff g \in \Theta(f)$.
7. Montrer que $\max(f, g) \in \Theta(f + g)$.
8. Qu’en est-il de $\min(f, g)$?

Exercice 2 :

On considère ici des algorithmes prenant un paramètre entier n .

1. On suppose que l’algorithme **A**(n) se décompose en deux parties consécutives, de complexité (en temps) respective $f(n)$ et $g(n)$. Quelle est sa complexité totale?
2. On suppose que l’algorithme **B**(n) fait $f(n)$ tours de boucle, chacun ayant une complexité $g(n)$. Quelle est sa complexité?
3. On suppose que l’algorithme **C**(n) se décompose en deux étapes : un appel récursif à **C**($n-1$) (si $n > 0$), puis une opération de complexité $f(n)$. Quelle est sa complexité?

Exercice 3 :

Calculer la complexité des morceaux de code suivants :

1.

```
for i in range(n) :
    for j in range(n) :
        for k in range(n) :
            # instruction de coût constant
```
2.

```
for i in range(n) :
    for j in range(i) :
        # instruction de coût constant
```
3.

```
for i in range(n) :
    for j in range(i, n - i) :
        # instruction de coût constant
```

Exercice 4 : Compteurs

Soit T un tableau de n entiers compris entre 1 et m .

1. Écrire une fonction **compte** qui compte le nombre de fois que l’entier i apparaît dans le tableau T . Quelle est la complexité de cette fonction?

On souhaite maintenant remplir un tableau S de taille m de sorte que la i^{e} case du tableau contienne la réponse à la question précédente.

2. Proposer un premier algorithme utilisant la fonction **compte** pour remplir le tableau. Quelle est sa complexité?
3. Proposer un algorithme amélioré de complexité linéaire en n pour remplir le tableau S .

Exercice 5 : Manipulation de tableaux

Décrire des algorithmes pour les problèmes suivants et en donner la complexité en temps et en espace :

1. Trouver le plus grand élément d'un tableau T (d'entiers par exemple).
2. Déterminer le nombre d'éléments du tableau T inférieurs à une valeur x .
3. Déterminer si le tableau T est trié en ordre croissant.
4. Déterminer si le tableau T est *circulairement* trié (en ordre croissant), *i.e.* s'il existe un indice i tel que $T[i:] + T[:i]$ soit trié.