Université Paris 7 | Eléments d'Algorithmique
L2 | Année 2010-2011

TD 2

Performances asymptotiques

On s'intéresse a des ordre de grandeur, et a des termes dominant, l'idée étant que pour des entrées grandes, les effets des constantes et des termes d'ordre de grandeur inférieur sont négligeables.

La classe $O(g) = \{f, \text{ t.q. } \exists c \in \mathbb{N}, n_0 \geq 0, \forall n \geq n_0, |f(n)| \leq c.|g(n)|\}$ g est une borne supérieure asymptotique, elle peut décrire une approximation des cas les plus défavorables.

La classe $\Omega(g) = \{f, \text{ t.q. } \exists c \in \mathbb{N}^*, n_0 \geq 0, \forall n \geq n_0, c.|g(n)| \leq |f(n)|\}$ g est alors une borne inférieure asymptotique elle peut parler du cas le plus favorable.

La classe $\Theta(g) = \{f, \text{ t.q. } \exists c_1, c_2 \in \mathbb{N}^*, n_0 \geq 0, \forall n \geq n_0, 0 \leq c_1|g(n)| \leq |f(n)| \leq c_2|g(n)|\}$ On dira que g est une borne asymptotiquement approchée de f

rq : parfois on écrira des expressions de la forme $2n^2 + 3n + 1 = 2n^2 + \Theta(n)$

La classe $o(g) = \{f, \text{ t.q. } \lim_n f(n)/g(n) = 0\}$ On dira que f est négligeable devant g. Par exemple : $3n + 1 \in o(n^2)$

Rq : $f \in o(g) \Rightarrow f \in O(g)$

Exercice 1 Montrer que $0.5n^2 - 3n \in \Theta(n^2)$

Exercice 2 Montrer que $6n^3 \notin \Theta(n^2)$

Exercice 3 Montrer que $f \in \Theta(g)$ ssi $f \in O(g)$ et $f \in \Omega(g)$.

Exercice 4 Montrer que $max(f, g) \in \Theta(f + g)$

Exercice 5 Est ce que $2^{n+1} \in O(2^n)$? et $2^{2n} \in O(2^n)$

Exercice 6 La fonction 3^n est-elle dans $O(2^n)$?

Exercice 7 Montrer que $3n^2 + 2n \in O(n^2)$ en précisant les constantes et rangs associés dans la définition.

Exercice 8 Montrer en reprenant la définition de la classe Θ que si $f_1 \in \Theta(g_1)$ et que $f_2 \in \Theta(g_2)$ alors $f_1 * g_2 \in \Theta(f_2 * g_1)$

Exercice 9 Quel est la complexité du test qui répond à la question de savoir si un tableau de taille n est trié ?

Expliquez pourquoi tout algorithme de tri d'un tableau de taille n peut être transformé en algorithme $\Omega(n)$

Exercice 10
1. Soit h une fonction dont on cherche a estimer la complexité, peut-on avoir :
   - à la fois $h \in O(n^2)$ et $h \in O(n^3)$?
   - à la fois $h \in \Omega(n^2)$ et $h \in \Omega(n^3)$?
   - à la fois $h \in \Theta(n^2)$ et $h \in \Theta(n^3)$?
2. Est-il vrai que $f \in \Omega(g)$ si et seulement si $g \in O(f)$
3. Donner une classe de complexité θ claire de $f(n) = n(2 + sin^2\ n)$

Exercice 11 La fonction f est définie par $f(n) = 3n^2 + 1$ lorsque $2^n < n^{\log(n)}$ et $f(n) = n^3 + e^n$ sinon.

1. Quelle est la classe de complexité θ de f
2. Quelle est la classe de complexité θ de $(1/f)$
3. Montrer que lorsque $h \in \theta(g)$ alors $h \notin o(g)$

Exercice 12 Pour chacune des paires de fonctions suivantes $f(n)$ et $g(n)$, dire si f est $O(g)$, si g est $O(f)$ et si f est $\Theta(g)$:

a) $f(n) = ln(2n)$; $g(n) = ln(n^3)$

b) $f(n) = 2^{n/2}$; $g(n) = n^3$

Exercice 13

1. On sait qu'un boût de programme a une complexité en $O(f(n))$ et qu'il est exécuté dans une boucle qui est itérée $g(n)$ fois. Que peut on dire de la complexité totale de la boucle ?
2. On sait que deux boûts de programme ont une complexité en $O(f(n))$ et en $O(g(n))$ respectivement. Ils sont exécutés séquentiellement l'un après l'autre. Que peut on dire de la complexité totale ?