Partie 3

Outils pour l'analyse d'algorithmes

Analyse d'algorithmes

Dans cette partie du cours, on va voir des outils permettant *d'analyser des algorithmes* :

- c'est-à-dire d'évaluer leur coût, en termes de temps de calcul, nombres d'opérations, ou encore utilisation de la mémoire.

Matière :

- Sommations et notations asymptotiques
- Récurrences
- Fonctions génératrices (pour la résolution de récurrence et le dénombrement)

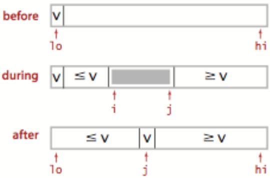
Sources :

- MCS
- R. Sedgewick et P. Flagolet, *Analysis of Algorithms*, Addison-Wesley, 1995. `http://aofa.cs.princeton.edu/`.
- J. L. Gross, *Combinatorial methods with computer applications*, Chapman & Hall, 2008.

Exemple introductif : quicksort

```
PARTITION(A, lo, hi)
  i = lo ; j = hi + 1 ; v = A[lo]
  while (true)
        repeat i = i + 1 until A[i] >= v
        repeat j = j - 1 until A[j] <= v
        if (i >= j)
              break
        swap(A[i], A[j])
  swap(A[lo], A[j]) ;
  return j
```

```
QUICKSORT(A, lo, hi)
  if begin < end
        q = PARTITION(A, lo, hi)
        QUICKSORT(A, lo, q - 1)
        QUICKSORT(A, q + 1, hi)
```



Quicksort partitioning overview

http://algs4.cs.princeton.edu

Approche scientifique pour l'analyse d'algorithmes

Pour analyser un algorithme : *Sur papier*

- Identifier une *opération abstraite* au cœur de l'algorithme.
- Développer un *modèle des entrées* de l'algorithme.
- Déterminer la fréquence d'exécution C_N de l'opération pour une entrée de taille N.
- Faire l'hypothèse que le coût de l'algorithme est $\sim aC_N$ où a est une constante.

Validation du modèle : *Sur ordinateur*

- Développer un générateur d'entrées selon le modèle
- Calculer a en exécutant l'algorithme pour des entrées larges
- Vérifier le résultat sur des entrées encore plus larges
- Valider le modèle d'entrée en testant l'algorithme sur une application réelle.

Quicksort

```
PARTITION(A, lo, hi)
i = lo ; j = hi + 1 ;v = A[lo]
while (true)
        repeat i = i + 1 until A[i] >= v
        repeat j = j − 1 until A[j] <= v
        if (i >= j)
                break
        swap(A[i], A[j])
swap(A[lo], A[j]) ;
return j
```

```
QUICKSORT(A, lo, hi)
if begin < end
        q = PARTITION(A, lo, hi)
        QUICKSORT(A, lo, q − 1)
        QUICKSORT(A, q + 1, hi)
```

- Opération de base : comparaison
- Modèle d'entrée :
  - Tableau A ordonné aléatoirement
  - Toutes les valeurs de A sont différentes
- Hypothèse : temps de calcul est $\sim aC_N$ où a est une constante et C_N est le nombre de comparaisons.

Modèle mathématique

Etant donné le modèle :

- Nombre de comparaisons pour le partitionnement : $N+1$
- Probabilité que le pivot soit à la position k : $1/N$
- Tailles des sous-tableaux dans ce cas-là : $k-1$ et $N-k$
- Les sous-tableaux sont aussi triés aléatoirement

Le nombre *moyen* de comparaisons utilisées par le quicksort est donné par la récurrence suivante :

$$C_N = N + 1 + \sum_{k=1}^{N} \frac{1}{N}(C_{k-1} + C_{N-k})$$

Essayons de dériver une formulation analytique de cette récurrence (voir *chapitre 7*).

Formulation analytique

$$C_N = N + 1 + \sum_{k=1}^{N} \frac{1}{N}(C_{k-1} + C_{N-k})$$

Par symétrie :

$$C_N = N + 1 + \frac{2}{N} \sum_{k=1}^{N} C_{k-1}$$

En multipliant par N :

$$NC_N = N(N+1) + 2 \sum_{k=1}^{N} C_{k-1}$$

En soustrayant la même formule pour $N - 1$:

$$NC_N - (N-1)C_{N-1} = 2N + 2C_{N-1}$$

En rassemblant les termes :

$$NC_N = (N+1)C_{N-1} + 2N$$

On divise par $N(N+1)$:

$$\frac{C_N}{N+1} = \frac{C_{N-1}}{N} + \frac{2}{N+1}$$

Téléscopage :

$$\begin{aligned} \frac{C_N}{N+1} &= \frac{C_{N-1}}{N} + \frac{2}{N+1} = \frac{C_{N-2}}{N-1} + \frac{2}{N} + \frac{2}{N+1} \\ &= \frac{C_1}{2} + \frac{2}{3} + \ldots + \frac{2}{N} + \frac{2}{N+1} \end{aligned}$$

Simplification :

$$C_N = 2(N+1)\sum_{k=1}^{N} \frac{1}{k} - 2N$$

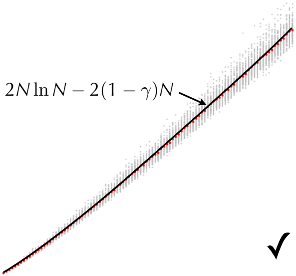
Approximation de la somme (*voir chapitre 6*)

$$C_N \sim 2N \ln N - 2(1-\gamma)N,$$

où $\gamma = 0.57721$.

Validation du résultat

Comparaison du modèle avec les valeurs réelles mesurées :



http://aofa.cs.princeton.edu/

- 1 point gris= 1 essai sur un tableau aléatoire
- 1 point rouge=moyenne pour chaque N

Limitations de l'approche scientifique

Le modèle peut ne pas être réaliste

- Pour le quicksort, on peut randomiser le tableau d'entrée avant d'appliquer l'algorithme pour se mettre dans les conditions du modèle d'entrée

Les maths peuvent être trop difficiles

- L'objectif des prochains cours est de vous donner quelques outils de base pour faire ce genre d'analyse.

Chapitre 6

Sommations et comportements asymptotiques

Plan

1. Sommations
   - Définitions
   - Preuve d'une solution analytique
   - Trouver une solution analytique
   - Approximation par intégration

2. Notations asymptotiques
   - $\sim$, o, et w
   - O, Ω et Θ
   - Démonstrations et remarques

Sources : MCS (chapitre 13), Gross (chapitre 3).

Sommations

Définition : Soit une suite $x_i (i \in \mathbb{Z})$. La sommation $\sum_{i=a}^{b} x_i$ pour $a, b \in \mathbb{Z}$ est définie récursivement par :

- $\sum_{i=a}^{b} x_i = 0$ si $b < a$ *(cas de base)*
- $\sum_{i=a}^{b} x_i = x_a$ si $b = a$ *(cas de base)*
- $\sum_{i=a}^{b} x_i = \left(\sum_{i=a}^{b-1} x_i\right) + x_b$ si $b > a$ *(cas inductif)*

Définition : Soit une suite de réels x_i, $i \in \mathbb{N}$, la *série* de *terme général* x_i est la suite de *sommes partielles*

$$\sum_{i=0}^{n} x_i \ (n \in \mathbb{N}).$$

Etant donné une série, on notera S_n la n-ème somme partielle $\sum_{i=0}^{n} x_i$. La suite des sommes partielles peut être définie récursivement :

- $S_0 = x_0$
- $S_n = S_{n-1} + x_n$ pour $n > 0$

Sommations

Les sommations apparaissent fréquemment dans le cadre de l'analyse d'algorithme et de la résolution de récurrences.

Objectif de ce chapitre : dériver des solutions analytiques à des sommations, et en particulier aux éléments d'une suite de somme partielle.

- Définition : Une *solution analytique* est une expression mathématique qui peut être évaluée à l'aide d'un nombre constant d'opérations de base (addition, multiplication, exponentiation, etc.).

But : simplifier l'évaluation des sommations pour prédire/étudier les performances d'un algorithme.

Preuve d'une solution analytique

Une solution analytique se prouve généralement facilement par induction.

Exemple : Série géométrique :

Théorème : Pour tous $n \geq 1$ et $z \neq 1$, on a

$$\sum_{i=0}^{n-1} z^i = \frac{1 - z^n}{1 - z}.$$

Démonstration : La preuve fonctionne par induction.
$P(n) = " \sum_{i=0}^{n-1} z^i = \frac{1-z^n}{1-z} "$
Cas de base $(n = 1)$: $P(1)$ est vérifié
Cas inductif $(n > 1)$: Si $P(n)$ est vérifié, on peut écrire :

$$\sum_{i=0}^{n} z^i = \sum_{i=0}^{n-1} z^i + z^n = \frac{1 - z^n}{1 - z} + z^n = \frac{1 - z^n + z^n - z^{n+1}}{1 - z} = \frac{1 - z^{n+1}}{1 - z}$$

$\square$

(Exercice : montrer que $\sum_{i=0}^{n} i^2 = \frac{(2n+1)(n+1)n}{6}$*)*

Sommes infinies

Définition : $\displaystyle\sum_{i=0}^{\infty} z_i = \lim_{n\to\infty} \sum_{i=0}^{n} z_i.$

Théorème : Si $|z| < 1$, alors $\displaystyle\sum_{i=0}^{\infty} z^i = \frac{1}{1-z}.$

Démonstration :

$$
\begin{aligned}
\sum_{i=0}^{\infty} z^i &= \lim_{n\to\infty} \sum_{i=0}^{n} z^i \\
&= \lim_{n\to\infty} \frac{1-z^{n+1}}{1-z} \\
&= \frac{1}{1-z}.
\end{aligned}
$$

□

Trouver une solution analytique

Si prouver une solution analytique est aisé, imaginer cette solution l'est moins.

Différentes techniques génériques existent :

- Dériver cette solution de la solution analytique d'une autre série (par exemple par dérivation ou intégration),
- Méthode de perturbation,
- Par identification paramétrique.
- ...

Variantes des séries géométriques

Théorème : Pour tous $n \geq 0$ et $z \neq 1$, on a

$$\sum_{i=0}^{n} iz^i = \frac{z - (n+1)z^{n+1} + nz^{n+2}}{(1-z)^2}.$$

Démonstration : On a

$$\sum_{i=0}^{n} iz^i = z \cdot \sum_{i=0}^{n} iz^{i-1} = z \cdot \left(\frac{d}{dz} \sum_{i=0}^{n} z^i \right) = z \cdot \left(\frac{d}{dz} \frac{1-z^{n+1}}{1-z} \right).$$

En développant, on obtient

$$\begin{aligned} & z \cdot \left(\frac{d}{dz} \frac{1-z^{n+1}}{1-z} \right) \\ = \quad & z \cdot \left(\frac{-(n+1)z^n(1-z) - (-1)(1-z^{n+1})}{(1-z)^2} \right) \end{aligned}$$

$$
\begin{aligned}
&= z \cdot \left(\frac{-(n+1)z^n + (n+1)z^{n+1} + 1 - z^{n+1}}{(1-z)^2} \right) \\
&= z \cdot \left(\frac{1 - (n+1)z^n + nz^{n+1}}{(1-z)^2} \right) \\
&= \frac{z - (n+1)z^{n+1} + nz^{n+2}}{(1-z)^2}.
\end{aligned}
$$

$\square$

Corollaire : Si $|z| < 1$, alors $\displaystyle\sum_{i=0}^{\infty} iz^i = \frac{z}{(1-z)^2}$.

Autre variante : En intégrant les deux côtés de $\sum_{i=0}^{\infty} z^i = \frac{1}{1-z}$ (de 0 à x), on peut obtenir :

$$\sum_{j=1}^{\infty} \frac{x^j}{j} = -\ln(1-x).$$

Méthode de perturbation

Soit S_n la n-ème somme partielle de la série de terme général x_i. Par définition, on a

$$S_n + x_{n+1} = x_0 + \sum_{i=1}^{n+1} x_k \; (= S_{n+1})$$

Si on peut exprimer le membre de droite comme une fonction de S_n, on peut obtenir une solution analytique en résolvant l'équation pour S_n.

Exemple : Pour la série géométrique $S_n = \sum_{i=0}^{n-1} z^i$:

$$S_{n+1} = S_n + z^n = z^0 + \sum_{i=1}^{n} z^i = 1 + z \sum_{i=0}^{n-1} z^i = 1 + zS_n$$

D'où, on tire immédiatement :

$$S_n = \frac{1 - z^n}{1 - z}$$

Un autre exemple

Problème[6] : Dériver une solution analytique de $S_n = \sum_{k=0}^{n} k2^k$.

Solution : Par la méthode de perturbation :

$$\begin{aligned}
S_n + (n+1)2^{n+1} &= 0 \cdot 2^0 + \sum_{k=1}^{n+1} k2^k = \sum_{k=1}^{n+1} k2^k \\
&= \sum_{k=0}^{n} (k+1)2^{k+1} \\
&= \sum_{k=0}^{n} k2^{k+1} + \sum_{k=0}^{n} 2^{k+1} \\
&= 2\sum_{k=0}^{n} k2^k + 2\sum_{k=0}^{n} 2^k \\
&= 2S_n + 2(2^{n+1} - 1) \\
\Rightarrow S_n &= (n-1)2^{n+1} + 2
\end{aligned}$$

[6] Cette somme apparaît dans l'analyse du tri par tas (voir INFO0902).

Perturbation indirecte

Parfois, ça ne marche pas directement.

Exemple : $S_n = \sum_{k=0}^{n} k^2$

$$\begin{aligned}
S_n + (n+1)^2 &= 0^2 + \sum_{k=1}^{n+1} k^2 \\
&= \sum_{k=0}^{n} (k+1)^2 = \sum_{k=0}^{n} (k^2 + 2k + 1) \\
&= \sum_{k=0}^{n} k^2 + 2\sum_{k=0}^{n} k + \sum_{k=0}^{n} 1 \\
&= S_n + 2\sum_{k=0}^{n} k + (n+1) \\
\Rightarrow \sum_{k=0}^{n} k &= \frac{n(n+1)}{2}
\end{aligned}$$

C'est correct mais ce n'est pas ce qu'on voulait calculer.

Perturbation indirecte

Dans ce cas, appliquer la perturbation à la suite $n \cdot x_n$ peut fonctionner :

Exemple : $S_n = \sum_{k=0}^{n} k \cdot k^2 = \sum_{k=0}^{n} k^3$

$$\begin{aligned}
S_n + (n+1)^3 &= 0^3 + \sum_{k=1}^{n+1} k^3 \\
&= \sum_{k=0}^{n} (k+1)^3 = \sum_{k=0}^{n} (k^3 + 3k^2 + 3k + 1) \\
&= \sum_{k=0}^{n} k^3 + 3\sum_{k=0}^{n} k^2 + 3\sum_{k=0}^{n} k + \sum_{k=0}^{n} 1 \\
&= S_n + 3\sum_{k=0}^{n} k^2 + 3\frac{n(n+1)}{2} + (n+1) \\
\Rightarrow \sum_{k=0}^{n} k^2 &= \frac{2(n+1)^3 - 3n(n+1) - 2(n+1)}{6} = \frac{(2n+1)(n+1)n}{6}
\end{aligned}$$

Par identification

On fait une hypothèse sur la forme de la solution et on identifie les paramètres en prenant quelques valeurs.

Exemple : $\sum_{i=1}^{n} i^2 = \frac{(2n+1)(n+1)n}{6}$.

- Supposer que la somme est un polynôme de degré 3 (car somme$\sim$intégration)

$$\sum_{i=1}^{n} i^2 = an^3 + bn^2 + cn + d$$

- Identifier les constantes a, b, c, d à partir de quelques valeurs de la somme
- Prouver sa validité par induction (!)

Série harmonique

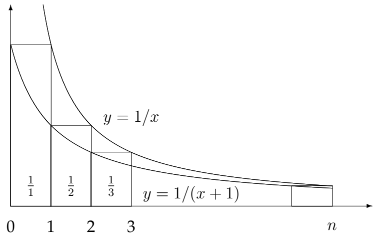
Complexité moyenne du quicksort (voir transp. 302) :

$$C_N = 2(N+1)\sum_{k=1}^{N}\frac{1}{k} - 2N$$

Définition : $H_n = \frac{1}{1} + \frac{1}{2} + \ldots + \frac{1}{n}$ est la *série harmonique*. H_n est le n-ème nombre harmonique.

La série harmonique n'a pas de solution analytique (connue). Des bornes inférieures et supérieures peuvent cependant être déterminées par intégration.

Approximation par intégration



$$\begin{aligned}
\int_0^n \frac{1}{x+1}\,dx &\leq H_n \leq & 1 + \int_1^n \frac{1}{x}\,dx \\
[\ln(x+1)]_0^n &\leq H_n \leq & 1 + [\ln x]_1^n \\
\ln(n+1) &\leq H_n \leq & 1 + \ln(n)
\end{aligned}$$

Définition : Soient deux fonctions $f, g : \mathbb{R} \to \mathbb{R}$. On écrit $f(x) \sim g(x)$ ssi $\lim_{x\to\infty} f(x)/g(x) = 1$ (f et g sont *asymptotiquement équivalents*).

$$\ln(n+1) \leq H_n \leq 1 + \ln(n)$$

$$\Rightarrow H_n \sim \ln n$$

Nombres harmoniques

Une meilleure approximation existe :

$$H_n = \ln(n) + \gamma + \frac{1}{2n} + \frac{1}{12n^2} + \frac{\epsilon(n)}{120n^4},$$

où γ est la constante d'Euler $(0.57721\ldots)$ et $0 \leq \epsilon(n) \leq 1$.

Complexité du quicksort :

$$\begin{aligned} C_N &= 2(N+1)H_N - 2N \\ &\sim 2N \ln N - 2(1-\gamma)N \end{aligned}$$

Méthode d'intégration

La méthode d'intégration peut être appliquée pour approximer beaucoup d'autres séries.

Définition : Une fonction $f : \mathbb{R}^+ \to \mathbb{R}^+$ est *strictement croissante* si $x < y$ implique $f(x) < f(y)$ et est *monotone croissante* si $x < y$ implique $f(x) \leq f(y)$.

Théorème : Soit une fonction $f : \mathbb{R}^+ \to \mathbb{R}^+$ monotone croissante. On a :

$$\int_1^n f(x)dx + f(1) \leq \sum_{i=1}^{n} f(i) \leq \int_1^n f(x)dx + f(n).$$

Le théorème peut être adapté trivialement aux fonctions décroissantes.

Exercice : montrez que $\frac{2}{3}n^{3/2} + \frac{1}{3} \leq \sum_{i=1}^{n} \sqrt{i} \leq \frac{2}{3}n^{3/2} + \sqrt{n} - \frac{2}{3}$.

Sommes doubles

Généralement, il suffit d'évaluer la somme intérieure et puis la somme extérieure.
Exercice : montrez que $\sum_{n=0}^{\infty}(y^n \sum_{i=0}^{n} x^i) = \frac{1}{(1-y)(1-xy)}$

Quand la somme intérieure n'a pas de solution analytique, échanger les deux sommes peut aider.
Exemple :

$$\begin{aligned}
\sum_{k=1}^{n} H_k &= \sum_{k=1}^{n}\sum_{j=1}^{k}\frac{1}{j} \\
&= \sum_{j=1}^{n}\sum_{k=j}^{n}\frac{1}{j} \\
&= \ldots \\
&= (n+1)H_n - n
\end{aligned}$$

$$\begin{array}{cc|ccccccc}
 & & j & & & & & & \\
 & & 1 & 2 & 3 & 4 & 5 & \ldots & n \\
\hline
k & 1 & 1 & & & & & & \\
 & 2 & 1 & 1/2 & & & & & \\
 & 3 & 1 & 1/2 & 1/3 & & & & \\
 & 4 & 1 & 1/2 & 1/3 & 1/4 & & & \\
 & & \ldots & & & & & & \\
 & n & 1 & 1/2 & & \ldots & & & 1/n
\end{array}$$

Remarque sur les produits

Les mêmes techniques peuvent être utilisées pour calculer des produits en utilisant le logarithme :

$$\prod f(n) = \exp\left(\ln\left(\prod f(n)\right)\right) = exp\left(\sum \ln f(n)\right).$$

Permet de borner $n! = 1 \cdot 2 \cdot 3 \ldots (n-1) \cdot n$. Par la méthode d'intégration, on a :

$$n\ln(n) - n + 1 \leq \sum_{i=1}^{n} \ln(i) \leq n\ln(n) - n + 1 + \ln(n).$$

En prenant l'exponentielle :

$$\frac{n^n}{e^n} \leq n! \leq \frac{(n+1)^{(n+1)}}{e^n}$$

Stirling's formula :

$$n! \sim \sqrt{2\pi n}\left(\frac{n}{e}\right)^n.$$

Plan

Notations asymptotiques

Les notations asymptotiques permettent de caractériser une fonction $f(x)$ lorsque x est très grand.

On a déjà vu la notion d'équivalence asymptotique ($\sim$) :
Définition : Soient deux fonctions $f, g : \mathbb{R} \rightarrow \mathbb{R}$. On écrit $f(x) \sim g(x)$ ssi $\lim_{x\to\infty} f(x)/g(x) = 1$ (f et g sont *asymptotiquement équivalents*).

On peut définir deux notations supplémentaires :
Définitions : Soient deux fonctions $f, g : \mathbb{R} \rightarrow \mathbb{R}$.

- On écrit $f(x) = o(g(x))$ ssi $\lim_{x\to\infty} f(x)/g(x) = 0$. On dira que f est négligeable devant g asymptotiquement.
- On écrit $f(x) = w(g(x))$ ssi $\lim_{x\to\infty} g(x)/f(x) = 0$. On dira que f domine g asymptotiquement.

Exemples :

- $2n = o(n^2)$, $2n^2 \neq o(n^2)$
- $n^2/2 = w(n)$, $n^2/2 \neq w(n^2)$

Quelques propriétés des notations $\sim$, o et w

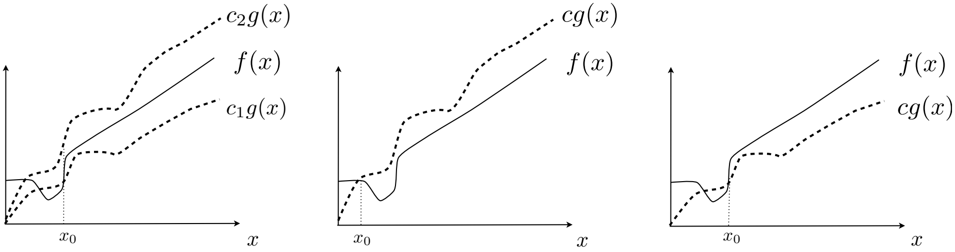
1. $f(x) = o(g(x))$ ssi **pour tout** $c > 0$, il existe un $x_0 \in \mathbb{R}$ tel que pour tout $x \geq x_0$, $|f(x)| \leq c|g(x)|$.
2. $f(x) = w(g(x))$ ssi **pour tout** $c > 0$, il existe un $x_0 \in \mathbb{R}$ tel que pour tout $x \geq x_0$, $|f(x)| \geq c|g(x)|$.
3. $f(x) = w(g(x))$ ssi $g(x) = o(f(x))$.
4. $f(x) \sim g(x)$ ssi $f(x) - g(x) = o(g(x))$.
5. $f(x) \sim g(x)$ ssi $f(x) = g(x) + h(x)$ pour une fonction $h(x) = o(g(x))$.
6. $x^a = o(x^b)$ pour tout $a < b$
7. $\log x = o(x^\epsilon)$ pour tout $\epsilon > 0$

(Exercices : démontrez ces propriétés)

Notations asymptotiques : O, Ω et Θ

Définitions : Soient $f : \mathbb{R} \to \mathbb{R}$ et $g : \mathbb{R} \to \mathbb{R}$ deux fonctions :

- On écrit $f(x) = O(g(x))$ s'il existe des constantes x_0 et $c > 0$ telles que $|f(x)| \leq c.|g(x)|$ pour tout $x \geq x_0$.
- $f(x) = \Omega(g(x))$ s'il existe des constantes x_0 et $c > 0$ telles que $|f(x)| \geq c.|g(x)|$ pour tout $x \geq x_0$.
- $f(x) = \Theta(g(x))$ s'il existe des constantes x_0, c_1 et $c_2 > 0$ telles que $c_1|g(x)| \leq |f(x)| \leq c_2|g(x)|$ pour tout $x \geq x_0$.



$f(x) = \Theta(g(x))$ $\quad$ $f(x) = O(g(x))$ $\quad$ $f(x) = \Omega(g(x))$

Quelques propriétés

1. $f(x) = \Omega(g(x)) \Leftrightarrow g(x) = O(f(x))$
2. $f(x) = \Theta(g(x)) \Leftrightarrow f(x) = O(g(x))$ et $f(x) = \Omega(g(x))$
3. $f(x) = \Theta(g(x)) \Leftrightarrow f(x) = O(g(x))$ et $g(x) = \Omega(f(x))$
4. $f(x) = o(g(x))$ ou $f \sim g \Rightarrow f(x) = O(g(x))$
5. $f(x) = O(g(x))$ et $g(x) = o(h(x)) \Rightarrow f(x) = o(h(x))$
6. $f(x) = O(g(x))$ et $g(x) = O(h(x)) \Rightarrow f(x) = O(h(x))$ (transitivité)
7. Si $f_1(x) = O(g_1(x))$ et $f_2(x) = O(g_2(x))$, alors
   $f_1(x) + f_2(x) = O(g_1(x) + g_2(x)) = O(\max\{g_1(x), g_2(x)\}$.
8. Si $f_1(x) = O(g_1(x))$ et $f_2(x) = O(g_2(x))$, alors
   $f_1(x)f_2(x) = O(g_1(x)g_2(x))$.

Les propriétés 6, 7 et 8 sont valable pour Θ, Ω, o et w.

NB : On peut faire une analogie entre ces notations et les comparateurs sur les réels : $o \rightarrow <$, $O \rightarrow \leq$, $\Theta \rightarrow =$, $\Omega \rightarrow \geq$, $w \rightarrow >$.

Démonstrations d'une relation asymptotique

Propriété : On a $5x + 100 = O(x)$.

Démonstration : On doit trouver des constantes x_0 et $c > 0$ telles que $|5x + 100| \leq cx$ pour tout $x \geq x_0$. Soient $c = 10$ et $x_0 = 20$. On a

$$|5x + 100| \leq 5x + 5x = 10x$$

pour tout $x \geq 20$. $\square$

Propriété : On a $x = O(x^2)$.

Démonstration : On doit trouver des constantes x_0 et $c > 0$ telles que $|x| \leq c \cdot x^2$ pour tout $x \geq x_0$. Soient $c = 1$ et $x_0 = 1$. On a

$$|x| \leq 1 \cdot x^2$$

pour tout $x \geq 1$. $\square$

Propriété : On a $x^2 \neq O(x)$.

Démonstration : Par l'absurde, supposons qu'il existe des constantes x_0 et $c > 0$ telles que

$$|x^2| \leq c \cdot x$$

pour tout $x \geq x_0$. On doit donc avoir

$$x \leq c$$

pour tout $x \geq x_0$, ce qui est impossible à satisfaire pour $x = \max(x_0, c + 1)$. $\square$

Remarques importantes

- On devrait écrire $f(x) \in O(g(x))$ plutôt que $f(x) = O(g(x))$
  - $O(g(x))$ n'est pas une fonction mais l'ensemble des fonctions $f(x)$ telles que $f(x) = O(g(x))$.
- Par abus de notation, on se permet d'écrire :

$$f(x) = g(x) + O(h(x))$$

qui signifie qu'il existe une fonction $i(x) = O(h(x))$ telle que :

$$f(x) = g(x) + i(x).$$

Exemples :

- $H_n = \ln n + \gamma + O(\frac{1}{n})$
- $2n^2 + \Theta(n) = \Theta(n^2)$
- $\sum_{i=1}^{n} O(i) = O(n^2)$ (mais $O(1) + O(2) + \ldots + O(n) = O(n^2)$ n'a pas de sens)