Fiche de TD4 : diviser pour régner

- Exo 1 -

Sur ALGO1 :

* **Équation de récurrence** : on note $t(n)$ le temps mis par l'algo sur une entrée n.

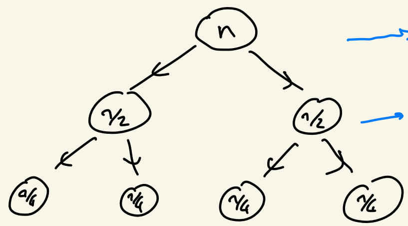
$$t(1) = O(1)$$

Pour $n > 1$ $\quad t(n) = 2\,t\left(\left\lceil \frac{n}{2} \right\rceil\right) + O(n^2)$

la complexité hors les appels récursifs (2 boucles imbriquées)

* **Estimation par arbre des appels récursifs** :

$t(n) \leq 2 \cdot t\left(\left\lceil \frac{n}{2} \right\rceil\right) + c \cdot n^2$



log n étapes

(l'ensemble de fois qu'on peut diviser n par 2).

$\longrightarrow \; c \cdot n^2$ (où c majore la somme des op. constante qui apparaissent dans l'algo).

$\longrightarrow \; 2 \times c\left(\frac{n}{2}\right)^2$ $\quad c\left(\frac{n}{2}\right)^2$: tps hors appel rec mis par l'algo avec $\frac{n}{2}$ comme paramètre

$\rightsquigarrow \; 4 \times c\left(\frac{n}{4}\right)^2$

$\rightsquigarrow \; 8 \times c\left(\frac{n}{8}\right)^2$

⋮

Somme : $t(n) \simeq \sum_{i=0}^{\log n} 2^i c \left(\frac{n}{2^i}\right)^2$

$$t(n) = cn^2 \sum_{i=0}^{\log n} 2^i \left(\frac{1}{2^i}\right)^2$$

$$= cn^2 \sum_{i=0}^{\log n} \frac{1}{2^i} = cn^2 \, \frac{1 - \left(\frac{1}{2}\right)^{\log n + 1}}{1 - \frac{1}{2}} = cn^2 \, \frac{1 - \left(\frac{1}{2}\right)^{\log n + 1}}{\frac{1}{2}}$$

$$= 2cn^2 \left(1 - \left(\tfrac{1}{2}\right)^{\log n + 1}\right) \leq 2cn^2.$$

On devrait avoir $t(n) = O(n^2)$.

* **Master Theorem** :

$$t(n) \leq \underset{a}{2}\, t\left(\left\lceil \frac{n}{\underset{b}{2}} \right\rceil\right) + O(n^{\overset{d}{2}})$$

$b^d = 2^2 = 4 > a = 2$ donc

$$t(n) = O(n^d) = O(n^2).$$