Preuve par recurrence

Comment prouver ?

$T(n) = \mathcal{O}(f(n))$

On va utiliser la recurrence !

1. On pose notre hypothese

$\mathcal{H}_n : T(n) \leq cf(n)$ Pour une constante $c > 0$ qu'on peut choisir aussi grand que necessaire.

2. On verifie un ou plusieurs cas de base

$\mathcal{H}_{n_0}, \mathcal{H}_{n_0+1}, \dots$

3. On suppose $\mathcal{H}_{n_0}, \mathcal{H}_{n_0+1}, \dots, \mathcal{H}_{n-1}$ vraies et on en deduit $\mathcal{H}_n$
4. On en deduit $\forall n \geq n_0, T(n) \leq cf(n)$ ce qui est la def de $T(n) = \mathcal{O}(f(n))$

Exercice 1

Recherche dichotomique

- $T(1) = \Theta(1)$
- $T(n) \leq T(\lfloor \frac{n}{2} \rfloor) + \Theta(1)$

Montrons que $T(n) = \mathcal{O}(logn)$

1. Hypothese: $\mathcal{H}_n : T(n) \leq clog_2 n$
2. On verifie les cas de base
   - $\mathcal{H}_1 : T(1) = \Theta(1) \leq c \times 0$ FAUX
   - $\mathcal{H}_2 : T(2) \leq T(1) + \Theta(1) = \Theta(1) \leq clog_2 2 = c$ VRAI

A partir de 2 ca a l'air de marcher…

3. $\forall n > 3$ supposons $\mathcal{H}_2, \mathcal{H}_3, \dots, \mathcal{H}_{n-1}$

$$T(n) \leq T(\lfloor \frac{n}{2} \rfloor) + \Theta(1)$$

$$T(n) \leq clog_2 \lfloor \frac{n}{2} \rfloor + \Theta(1)$$

$$T(n) \leq clog_2 \frac{n}{2} + \Theta(1)$$

$$T(n) \leq clog_2 n - clog_2 2 + \Theta(1)$$

$$T(n) \leq clog_2 n - (c - \Theta(1))$$

Donc $\mathcal{H}_n$ est verifiee $\forall n \geq 2$

On a montre $\forall n \geq 2, T(n) \leq clog_2 n$

Exercice 2

$\begin{cases} T(1) = \Theta(1) \\ T(n) = 2T(\lfloor \dfrac{n}{2} \rfloor) + \Theta(1) \end{cases}$

Mq: $T9N) = \mathcal{O}(n)$

Hypothese: $\mathcal{H}_n : T(n) \leq cn$ avec c aussi grand que necessaire.

- $\mathcal{H}_1 = \Theta(n) \leq c$ VRAI

$\forall n > 1$, supposons $\mathcal{H}_1, \ldots, \mathcal{H}_{n-1}$ Vraies

$T(n) = 2T(\lfloor \frac{n}{2} \rfloor) + \Theta(1)$
$T(n) \leq 2c\lfloor \frac{n}{2} \rfloor + \Theta(1) \leq cn + \Theta(1)$

On ne peut **pas** conclure que $T(n) \leq cn$

2^eme^ essai:

$\mathcal{H}'_n : T(n) \leq c(n-1)$

- $\mathcal{H}'_1 : T(1) = \mathcal{O}(1) \leq c \times 0$ FAUX
- $\mathcal{H}'_2 : T(2) = \mathcal{O}(1) \leq c$ VRAI

$\forall n \geq 3$ On suppose $\mathcal{H}'_2, \ldots, \mathcal{H}'_{n-1}$

$$T(n) = 2T(\lfloor \frac{n}{2} \rfloor) + \Theta(1)$$

$$T(n) \leq 2c(\lfloor \frac{n}{2} \rfloor - 1) + \Theta(1)$$

$$T(n) \leq cn - 2c + \Theta(1)$$

$$T(n) \leq c(n-1) - c + \Theta(1)$$

$$T(n) \leq c(n-1) - (c - \Theta(1))$$

Donc: $T(n) \leq c(n-1)$

On a montre:

$\forall n \geq 2, T(n) \leq c(n-1) \leq cn$
D'ou $T(n) = \mathcal{O}(n-1) = \mathcal{O}(n)$

Quick Select

Il cherche la valeur de rang $k \in [0, n - l[$ dans letableau $A[l \dots r - 1]$

```
def QuickSelect(A, l, r, k):
  if r - l == 1:
    return A[l]
  m = Partition(A, l, n)
  L = m - l
  if k < L:
    return QuickSelect(A, l, m, k)
  else
    return QuickSelect(A, m, r, k - L)
```

Complexite

$T(1) = \Theta(1)$

$$T(n) = \Theta(n) + \begin{cases} T(L) \\ ou \\ T(n-L) \end{cases}$$

On peut pas appliquer la recurrence.

Cas moyen

On suppose L tire uniformement dans $\{1, 2, \dots, n-1\}$ et on evalue la complexite en moyenne.

$$T(n) \leq \mathcal{O}(n) + \tfrac{1}{n-1} \sum_{L=1}^{n-1} T(max(L, n-L))$$

$$T(n) \leq \Theta(n) + \tfrac{2}{n-1} \sum_{L=\lfloor \frac{n}{2} \rfloor} {}^{n-1}T(L)$$

Montrons que $T(n) = \mathcal{O}(n)$ par recurrence.

$\mathcal{H}_n : T(n) \leq cn$

$\mathcal{H}_1 : T(1) = \Theta(1) \leq c$ VRAI

pour $n > 1$ on suppose
$mathcalH_1, \dots, \mathcal{H}_{n-1}$

$$T(n) \leq \Theta n + \tfrac{2}{n-1} \sum_{L=\lfloor \frac{n}{2} \rfloor}^{n-1} T(L)$$

$$T(n) \leq \Theta n + \tfrac{2}{n-1} \sum_{L=\frac{n}{2}}^{n-1} cL$$

$$T(n) \leq \Theta n + \tfrac{2c}{n-1} \sum_{i=\lfloor \frac{n}{2} \rfloor}^{n-1} i$$

Lemme: $\displaystyle\sum_{i=\lfloor \frac{n}{2} \rfloor}^{n-1} i \leq \frac{3}{8} n^2$

$T(n) \leq \Theta n + \frac{2c}{n-1} \times \frac{3}{8} n^2$
$T(n) \leq \Theta n + \frac{6c}{8(n-1)} \times n((n-1)+1)$
$T(n) \leq \Theta n + \frac{6}{8} cn + \frac{6c(n-1+1)}{8(n-1)} + \Theta(n)$
$T(n) \leq cn$

$\implies T(n) = \mathcal{O}(n)$