

# CHAP 1: PRACTICAL AND THEORETICAL FOUNDATIONS OF PROGRAMMING.

Règles de pb:

Problème  $\rightarrow$  Spécification  $\rightarrow$  Algorithme  $\rightarrow$  Software syst.

- Abstraction
- Diviser pour mieux régner
- Modularité.

$\rightarrow$  Mesure de l'efficacité:

- Empirique  
↳ Trop de paramètres variables

- Estimation mathématique.

Ex: Recherche dans une liste.

- $t_{min}$ : Meilleur des cas.
- $t_{max}$ : Pire cas.
- $t_{avg}$ :  $p_1 t_1 + p_2 t_2 + \dots + p_n t_n$   
probas de succès



trav souvent complexe  $\Rightarrow$  ANALYSE ASYMPTOTIQUE.

Def:  $O$ -nota°.

$$T(n) \in O(f(n)) \Leftrightarrow \exists c, n_0 \text{ constantes}, \forall n > n_0, T(n) \leq c \cdot f(n)$$

Def:  $\Omega$ -nota°.

$$T(n) \in \Omega(f(n)) \Leftrightarrow \exists c, n_0 \text{ constantes}, \forall n > n_0, T(n) \geq c \cdot f(n)$$

Def:  $\Theta$ -nota°.

$$T(n) \in \Theta(g(n)) \Leftrightarrow T(n) \in O(g(n)) \text{ et } T(n) \in \Omega(g(n))$$