Algorithmique « avancée »

CM1 : Algorithmique « avancée » et efficacité

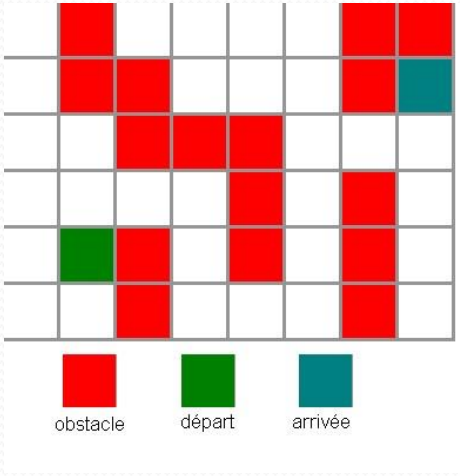
Mickaël Martin Nevot

V2.0.0

Algorithmique « avancée »

Efficacité d'un algorithme

- Particularités de l'ordinateur
- Performance en moyenne (**évaluation asymptotique**)
- **Complexité** :
  - L'évolution du nombre d'instructions de base en fonction de la quantité de données à traiter
  - Quantité de mémoire nécessaire pour effectuer les calculs



Complexité

- Comptabiliser le **nombre d'étapes d'un algorithme**
- Calculer dans **le pire des cas**
- Notation : $O(n)$ avec n étant le nombre d'étapes

```
// Complexité O(n)
for (int i = 0 ; i < n ; ++i) { … }

// Complexité O(n²) : avec m ≤ n
for (int i = 0 ; i < m ; ++i) {
    for (int j = 0 ; j < n ; ++j) { … }
}

// Complexité O(log(n))
… // Fonction Dichotomie
```

Complexité : comparatif

Notation	Complexité	Temps n = 10	Temps n = 10000	Temps n = 1000000	Exemple
$O(1)$	Constante	10 ns	10 ns	10 ns	Index tableau
$O(\log n)$	Logarithmique	10 ns	40 ns	60 ns	Dichotomie
$O(n)$	Linéaire	100 ns	100 µs	10 ms	Parcours liste
$O(n \log n)$	Quasi-linéaire	100 ns	400 µs	60 ms	Tri fusion
$O(n^2)$	Quadratique	1 µs	1 s	2.8 h	Tri par insertion
$O(n^3)$	Cubique	10 µs	2.7 h	316 ans	
$O\left(n^{\log n}\right)$	Quasi-polynomiale	100 ns	3.2 ans	10^{20} ans	
$O(e^n)$	Exponentielle	10 µs	…	…	DPFP
$O(n!)$	Factorielle	36 ms	…	…	Voyageur de commerce

Crédits

Auteur

Mickaël Martin Nevot
mmartin.nevot@gmail.com

Carte de visite électronique

Relecteurs

Cours en ligne sur : www.mickael-martin-nevot.com