

UNIVERSITÉ LIBRE DE BRUXELLES – FACULTÉ DES SCIENCES
DÉPARTEMENT D'INFORMATIQUE

Abdallah EL HUSSEIN, Sam NADIF, Clément SINON et Alexis SMEETS

La complexité

⚠ Attention
algorithme ≠ programme

L'unité de mesure principale des algorithmes : le nombre d'opérations

Un algorithme n'a pas de temps d'exécution propre. C'est juste une succession d'opérations, qu'elles soient exécutées lentement ou rapidement.

On ne mesure pas un algorithme en **temps d'exécution**, mais en **nombre d'opérations effectuées**.

La taille des données et le nombre d'opérations : la complexité

Généralement, plus les données à traiter sont importantes, plus il y a d'opérations à effectuer pour compléter un algorithme.

La relation entre le nombre d'opérations à effectuer et la taille des données définit la complexité d'un algorithme.

La taille des données est souvent représentée par la lettre **n**.

Algorithme

```
pour chaque élément i:
  faire : f(i) -----
pour chaque élément j :
  faire : g(i,j) -----
```

Nombre d'opérations

n fois

n x n fois

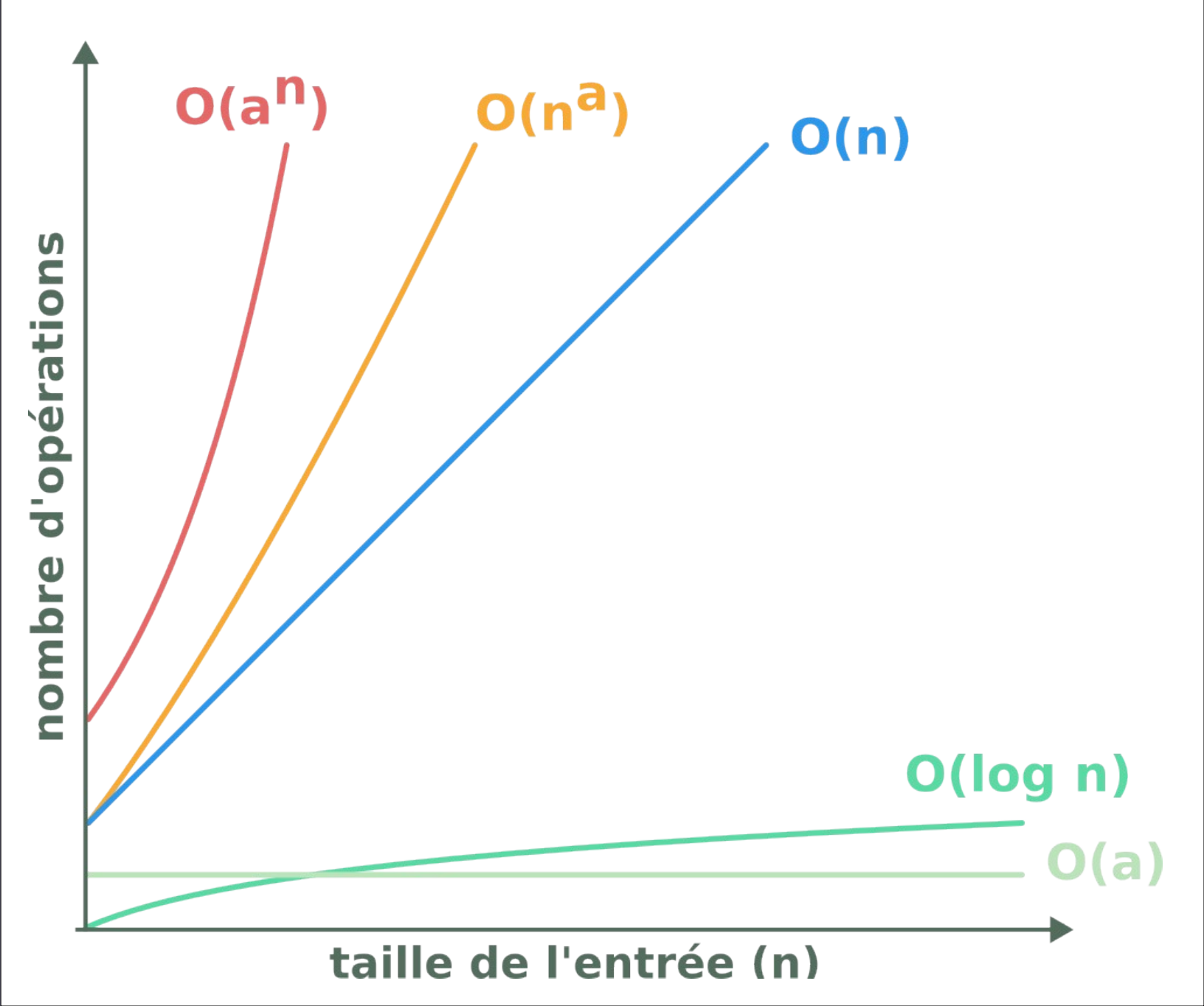
Complexité $n + n \times n = n + n^2$

Avec le big data : la complexité asymptotique

En informatique, on cherche à travailler avec des ensembles de données aussi grands que possibles. On s'intéresse à la complexité **lorsque la taille des données tend vers l'infini**.

Pour cela, on utilise la notation **Grand-O** : on oublie les détails et on ne conserve que les termes qui dominent l'expression lorsque la taille des données augmente.

$n + n^2 \rightarrow O(n^2)$



La complexité des algorithmes du
problème du rendu de monnaie

Les variables

La difficulté de résolution du problème du rendu de monnaie dépend de deux variables : **C** et **V**.

En euros, **C** = 15 (8 pièces et 7 billets) et **V** est exprimé en centimes.

Comme **C** reste constant d'une utilisation à une autre, on va s'intéresser ici à la complexité par rapport à **V**.

Voyons ce que cela donne avec notre exemple de **67 ¢**.

Deux variables	
Nombre de pièces / billets différents	C
Montant à rendre	V

Glouton

Un calcul par pièce.

$O(C) \rightarrow$ Constant sur V

Jusqu'à 15 opérations.

Naïf

Un arbre avec **C** embranchements par niveau et au maximum **V** niveaux.

$O(C^V) \rightarrow$ Exponentiel sur V

Jusqu'à 15⁶⁷ opérations !

Programmation Dynamique

(une amélioration du naïf)

C calculs simples pour chaque valeur de 1 à **V**.

$O(C \times V) \rightarrow$ Linéaire sur V

Jusqu'à 15x67 opérations.

6 282 237 584 682 227 977 202 510 791 050 497 017 916 354 192 124 117 616 913 281 381 130 218 505 859 375

1 005