

# Les arbres I : généralités

## Capacités attendues

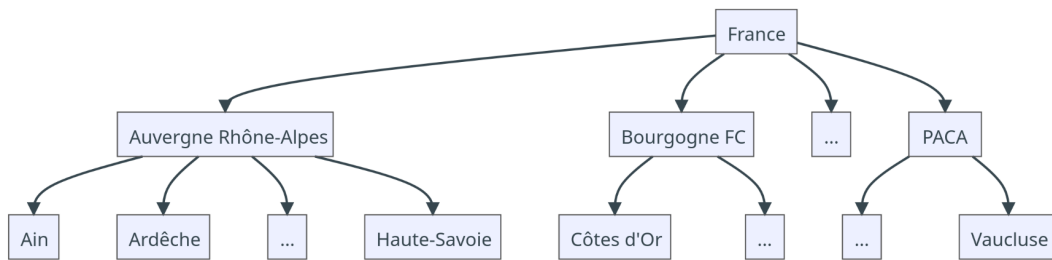
- ✓ Identifier des situations nécessitant une structure de données arborescente.

On a vu en étudiant leur complexité que les structures linéaires sont peu pratiques pour faire des recherches, des ajouts ou des suppressions de données ordonnées. Les structures arborescentes ou arbres, dont les utilisations en informatique sont multiples, sont plus adaptées à cela.

## 1 Exemples de structures arborescentes

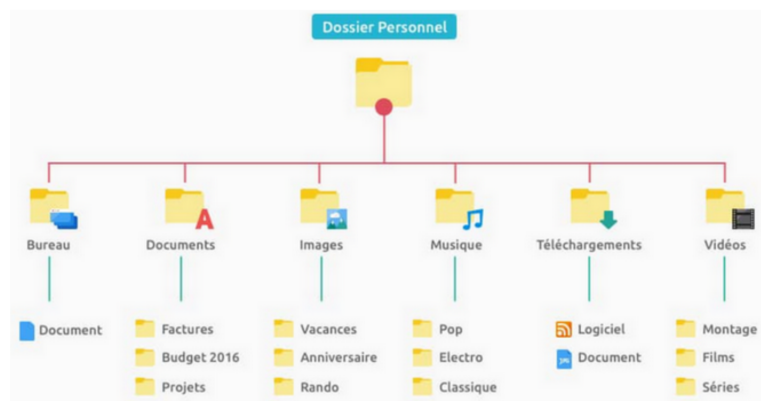
### 1.1 Géographie

**Exercice 1** Identifier, sur la représentation ci-dessous, à quoi correspondent les différentes lignes de l'arbre.



### 1.2 Arborescence de fichiers

**Exercice 2** Ci-dessous un exemple de stockage de fichiers et dossiers sur un ordinateur. Représentez-le de manière similaire à l'arbre du premier exemple :



## 1.3 Tournoi sportif

**Exercice 3** Représenter un tournoi à partir des quarts de finales jusqu'à la représentation du vainqueur en utilisant un arbre. Les quarts de finales opposent les équipes suivantes : France-Angleterre (vainqueur France), Maroc-Portugal (vainqueur Maroc), Pays-Bas-Argentine (vainqueur Argentine) et Croatie-Brésil (vainqueur Croatie). En demi-finale, la France bat le Maroc et l'Argentine la Croatie. Vous connaissez sûrement le résultat de la finale...

## 2 Définitions

### À retenir

En informatique, un arbre est une structure de données hiérarchique composée de **nœuds**.  
Chaque nœud d'un arbre est composé :

- **d'une étiquette (une valeur de type quelconque) ;**
- **d'un tableau de nœuds appelée « enfants » ou « sous-arbres ».**

Un arbre est déterminé par le seul nœud qui n'a pas de parent, appelé **racine**.

Un nœud qui n'a pas d'enfant est appelé **feuille**.

Les nœuds qui ont à la fois un parent et des enfants sont les **nœuds internes**.

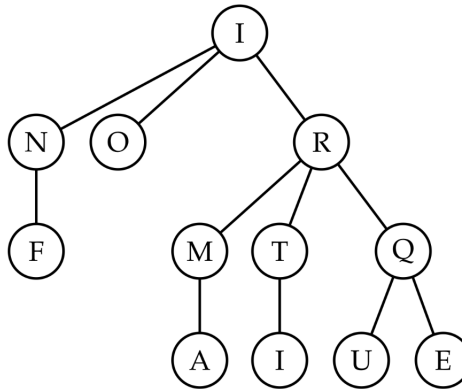
On représente un arbre en joignant le nœud racine à ses enfants par une **branche** ou **arête**, et en faisant de même récursivement pour chacun de ses enfants.

- La **taille** d'un arbre est **le nombre de nœuds qu'il contient**.
- La **profondeur** d'un nœud est **le nombre d'arêtes qui le joint à la racine**.
- La **hauteur** d'un arbre est **la plus grande profondeur de ses feuilles**.
- L'**arité d'un nœud** est **le nombre de ses enfants**.
- L'**arité d'un arbre** est **la plus grande arité de ses nœuds**.

**Représentation** : L'arbre informatique est visuellement représenté par un « arbre renversé ». La racine est généralement représentée tout en haut.

#### Exercice 4

On a représenté un arbre ci-dessous.



1. Quelles sont les feuilles de cet arbre ? Ses nœuds internes ?
2. Quelle est la taille de cet arbre ?
3. Quelle est l'arité du nœud d'étiquette "Q" ? Quelle est l'arité de cet arbre ?
4. Quelle est la profondeur du nœud d'étiquette "T" ? Quelle est la hauteur de cet arbre ?

#### Exercice 5

Dessiner un arbre de taille 9, étiqueté par des nombres de 0 à 8, de racine 0 et de hauteur 2.

#### Exercice 6

1. Quelle est la hauteur maximale d'un arbre de taille  $t$  ?
2. Quelle est la taille maximale d'un arbre  $n$ -aire (d'arité  $n$ ) de hauteur  $h$  ?
3. Quelle est la hauteur minimale d'un arbre binaire (c'est-à-dire un arbre d'arité 2) de taille  $t$  ?