

## 7.1

## Les arbres

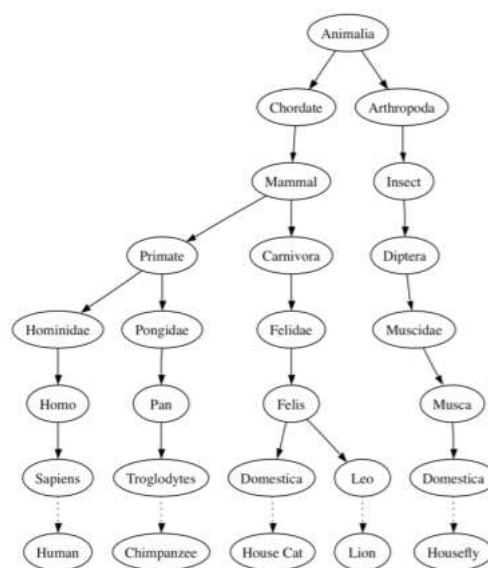
NSI TLE - JB DUTHOIT

## 7.1.1 Introduction

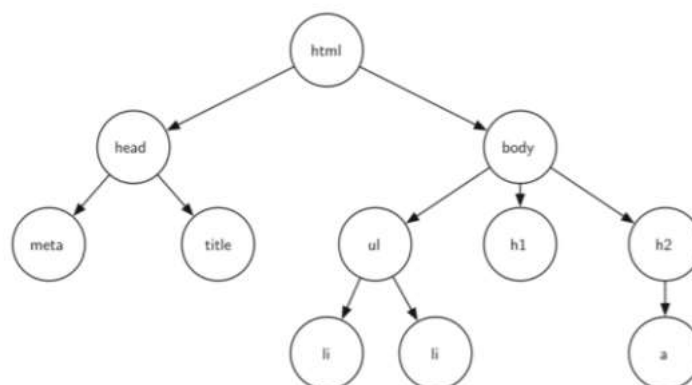
Il n'existe pas qu'une façon linéaire de représenter les données, comme les listes, les tableaux, les dictionnaires, les piles et les files. Nous pouvons également structurer les données de façon hiérarchique.

## 7.1.2 Exemples de situations où l'on rencontre des arbres

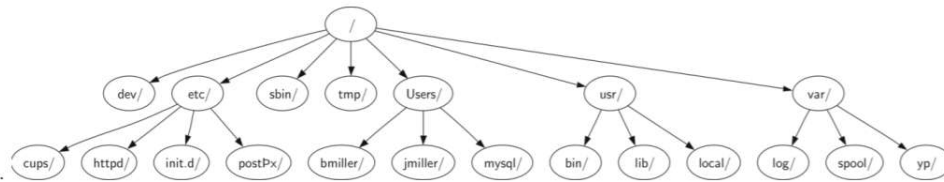
- En biologie



- Les balises d'une page web



- Les dossiers d'un ordinateur



### 7.1.3 Définition

#### Définition

Un **arbre** est une structure de données constituée de nœuds, qui peuvent avoir des enfants (et qui sont aussi des nœuds)

#### Définition

Le sommet de l'arbre est appelé **racine**.

#### Définition

Un nœud qui ne possède pas d'enfant est appelé une **feuille**.

#### Définition

Les nœuds autres que la racine et les feuilles sont appelés **nœuds internes**.

#### Définition

Une **branche** est une suite finie de nœuds consécutifs de la racine vers une feuille.

#### Remarque

Un arbre a donc autant de feuilles que de branches !

#### Définition

l'**arité** d'un arbre est le nombre maximal d'enfants qu'un nœud peut avoir.

#### Définition

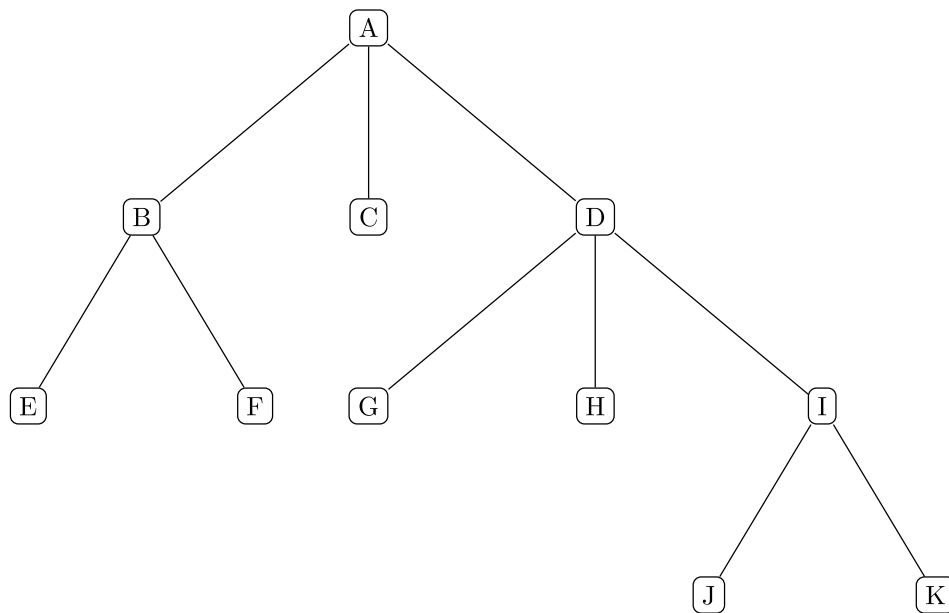
La **taille** d'un arbre est le nombre de nœuds qui le compose.

#### Définition

La **hauteur** d'un arbre est la profondeur à laquelle il faut descendre pour trouver la feuille la plus éloignée de la racine.

**Exercice 7.88**

On considère l'arbre ci-dessous.



Répondre aux questions suivantes :

- Nombre de nœuds ?
- Nombre de racine ? Laquelle ?
- Nombre de feuilles ? Lesquelles ?
- Nombre de branches ?
- Nombre de nœuds internes ?
- Quel est son arité ?
- Quelle est sa taille ?
- Quelle est sa hauteur ?

**Remarque**

| En informatique, les arbres poussent vers le bas :-)

**7.1.4 Représentation en Python d'un arbre, appelé aussi arborescence****Avec une classe**

Pour représenter un arbre (une arborescence) en Python, on peut utiliser des objets, comme pour les listes chaînées.

L'objet de la classe contient deux attributs : un attribut valeur (dans lequel on stocke une valeur quelconque, appelée étiquette) et un attribut fils dans lequel on stocke les fils sous la forme d'un tableau.

**Exercice 7.89**

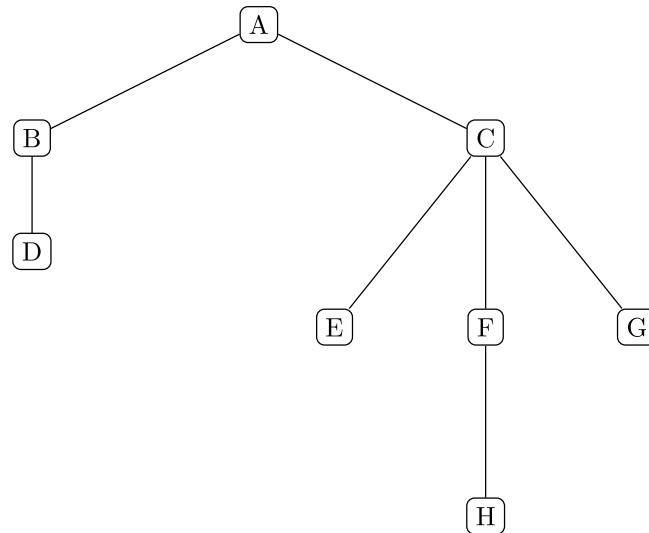
| Construire la classe Noeud. Pour les feuilles, on mettra [] pour le fils\*\*\*

**Exercice 7.90**

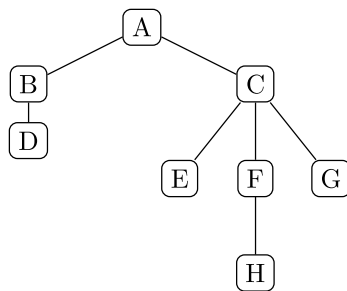
Construire l'arbre donné en exemple plus haut.

**Exercice 7.91**

Construire l'arbre ci-dessous :

**Exercice 7.92**

Cr  er une fonction r  cursive **represente(arbre,p=0)** qui permet un affichage d'un arbre comme ceci :



```

  A
  - B
  -- D
  - C
  -- E
  -- F
  --- H
  -- G
  
```

Repr  sentation possible d'un arbre. Les tirets indiquent la profondeur

\*\*\*

Avec un dictionnaire