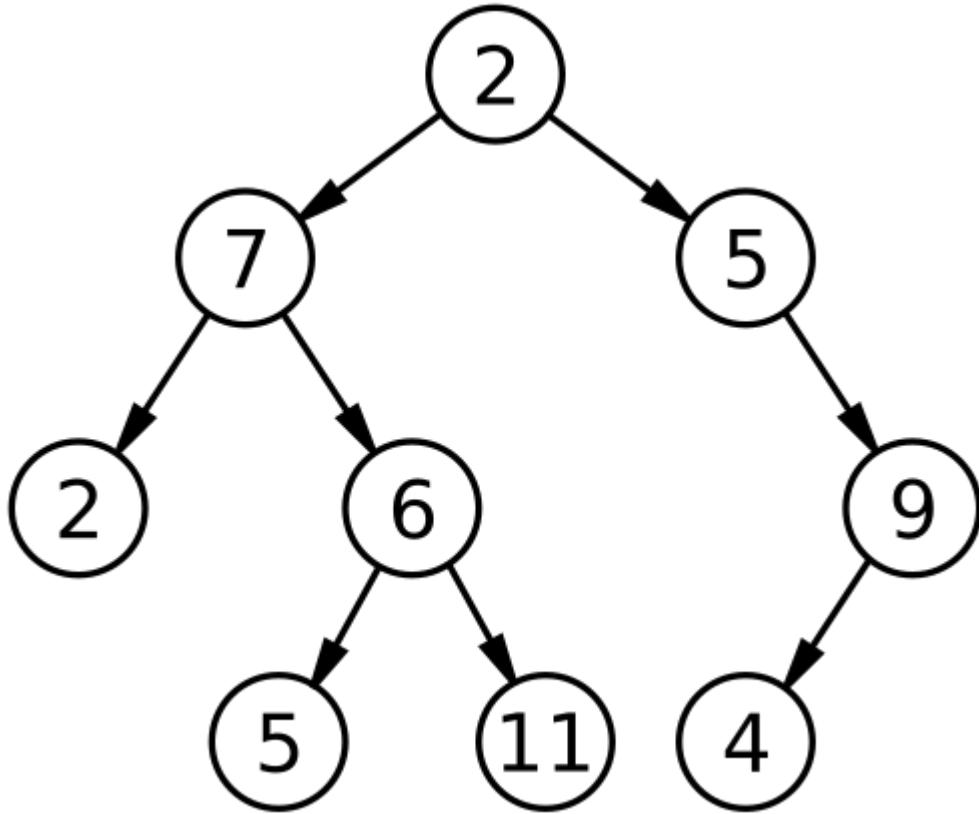


2. Arbres binaires

- ◆ Définition

Un **arbre binaire** est un arbre dans lequel chaque nœud a **au plus deux enfants** (gauche et droit).



Implémentation récursive (Python)

```
from __future__ import annotations

class ArbreBinaire:
    """Structure de donnée d'arbre binaire"""

    def __init__(self, étiquette: str, gauche: ArbreBinaire, droit: ArbreB
                 self.étiquette = étiquette
                 self.gauche = gauche
                 self.droit = droit
```

Chaque nœud contient :

- une **étiquette**,
- un **sous-arbre gauche**,

- un **sous-arbre droit**.

Parcours d'un arbre binaire

Il existe plusieurs façons de parcourir ou visiter les noeuds :

Parcours en largeur d'abord

Visite les noeuds **de haut en bas** puis **de gauche à droite** (comme la lecture d'un texte).

Parcours en profondeur d'abord

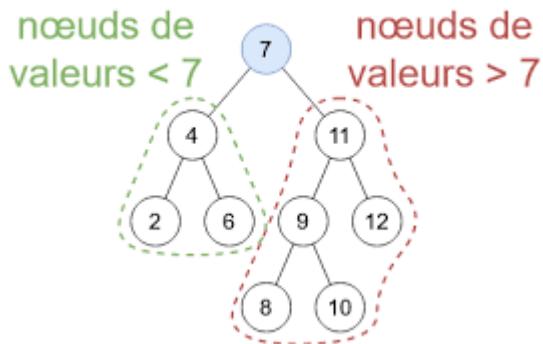
Type	Ordre
Préfixe (ou préordre)	Visite le noeud → gauche → droit.
Infixe (ou en-ordre)	Gauche → noeud → droit.
Postfixe (ou postordre)	Gauche → droit → noeud

3 Arbres binaires de recherche (ABR)

- ◆ Définition

Un **arbre binaire de recherche** est un **arbre binaire** où :

- dans le **sous-arbre gauche, toutes les valeurs sont inférieures** à la racine,
- dans le **sous-arbre droit, toutes les valeurs sont supérieures ou égales** à la racine.



Propriétés

- Un ABR permet des **recherches rapides**, car les valeurs y sont **ordonnées**.