



Université Sultan Moulay Slimane
Faculté Polydisciplinaire **Khouribga**



Sciences Mathématiques et Informatique

Structures de Données

Chapitre 6 : Structures Arboréscentes (Partie 1)

Pr. Ibtissam Bakkouri

i.bakkouri@usms.ma

Année Universitaire : **2022/2023**

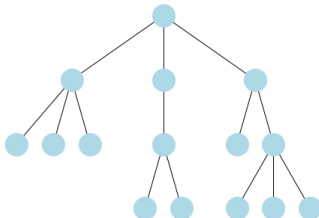
Plan

- 1 Introduction
- 2 Terminologie
- 3 Utilisations des arbres
- 4 Arbres Binaires

Introduction

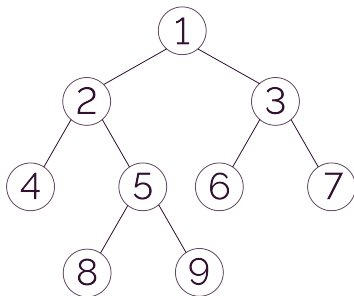
C'est quoi une structure arboréscente ?

Les structures arboréscentes sont des structures de données utilisées pour organiser et stocker des données sous forme d'arbre. Un arbre est une structure de données hiérarchique composée de nœuds reliés les uns aux autres par des arêtes appelées branches. Le nœud en haut de l'arbre est appelé la racine et les nœuds sans enfants sont appelés les feuilles.



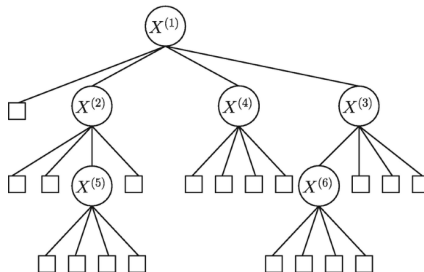
Introduction

Les structures arborées sont largement utilisées en informatique pour stocker des données hiérarchiques, telles que des fichiers et des répertoires dans un système de fichiers, des éléments dans une page Web, des catégories et des sous-catégories dans un système de classification, etc.



Introduction

Les arbres sont souvent représentés graphiquement sous forme de diagrammes arborescents, où chaque nœud représente un objet ou une entité, et les branches représentent les relations entre les objets. Les structures arborées peuvent être utilisées pour effectuer des opérations telles que la recherche, l'insertion, la suppression et la mise à jour de données.



Terminologie

La terminologie associée aux structures arborées inclut :

- **Nœud** : un élément de base d'un arbre, qui représente une entité ou un objet et qui peut contenir une valeur ou des références à d'autres nœuds.
- **Arête (ou branche)** : une connexion entre deux nœuds qui représente une relation hiérarchique entre eux.
- **Racine** : le nœud supérieur d'un arbre, qui n'a pas de parent et à partir duquel tous les autres nœuds sont accessibles.
- **Feuille (ou nœud terminal)** : un nœud sans enfants, qui représente une entité de base dans l'arbre.
- **Sous-arbre** : un arbre qui est une partie d'un arbre plus grand, composé d'un nœud et de tous ses descendants.

Terminologie

- **Profondeur** : la longueur du chemin à partir de la racine jusqu'à un nœud donné.
- **Hauteur** : la longueur du plus long chemin d'un nœud jusqu'à une feuille.
- **Niveau** : la profondeur d'un nœud, plus un.
- **Arbre binaire** : un arbre dans lequel chaque nœud a au plus deux enfants.
- **Arbre binaire de recherche** : un arbre binaire dans lequel chaque nœud a une valeur qui est supérieure à toutes les valeurs dans son sous-arbre gauche et inférieure à toutes les valeurs dans son sous-arbre droit.

Utilisations des arbres

Les structures arborées sont utilisées dans de nombreux domaines pour représenter des données de manière hiérarchique. Voici quelques exemples d'utilisation des structures arborées :

- **Représentation de données** : Les arbres sont souvent utilisés pour représenter des données hiérarchiques, telles que les fichiers et les répertoires sur un système de fichiers, les catégories et sous-catégories d'un site web, les relations de parenté dans un arbre généalogique, etc.
- **Algorithmes de recherche** : Les arbres sont utilisés dans de nombreux algorithmes de recherche, tels que l'algorithme de recherche binaire, l'arbre de décision, etc.
- **Représentation de langages** : Les arbres sont utilisés dans la théorie des langages formels pour représenter la structure de phrases et de mots.

Utilisations des arbres

- **Traitement de texte** : Les arbres sont souvent utilisés pour représenter la structure d'un document, en utilisant des éléments tels que des paragraphes, des listes, des en-têtes, etc.
- **Intelligence artificielle** : Les arbres sont utilisés dans de nombreuses applications d'intelligence artificielle, tels que les arbres de décision pour la classification, les arbres de syntaxe pour l'analyse grammaticale, etc.
- **Infographie** : Les arbres sont utilisés pour représenter des données sous forme de diagrammes arborescents, tels que les diagrammes de flux de processus, les diagrammes de structure organisationnelle, etc.

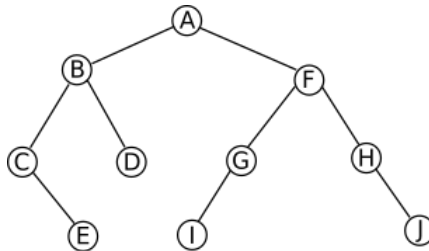
Utilisations des arbres

- **Gestion de projet** : Les arbres sont souvent utilisés pour représenter les tâches d'un projet sous forme d'arborescence, en utilisant des éléments tels que des tâches, des sous-tâches, des jalons, etc.
- **Réseaux informatiques** : Les arbres sont utilisés dans les réseaux informatiques pour représenter la hiérarchie de la structure des réseaux, tels que les réseaux LAN, WAN et Internet.

Arbres Binaires

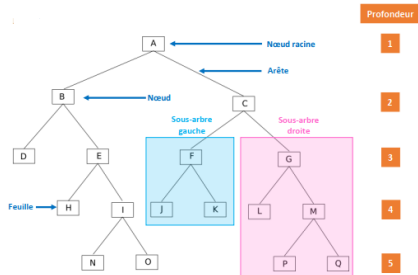
C'est quoi un Arbre Binaire ?

Un arbre binaire est une structure de données arborescente dans laquelle chaque nœud a au plus deux fils, appelés fils gauche et fils droit. Un nœud qui n'a pas de fils est appelé une feuille. L'arbre commence par un nœud appelé la racine, à partir de laquelle on peut atteindre tous les autres nœuds en descendant le long des fils.



Arbres Binaires

Chaque nœud peut contenir une donnée, et les données sont organisées de manière à ce que la donnée dans un nœud soit plus petite que toutes les données dans son sous-arbre droit et plus grande que toutes les données dans son sous-arbre gauche. Cette propriété est appelée la propriété d'ordre des arbres binaires de recherche.



Types d'arbres binaires

Il existe plusieurs types d'arbres binaires, chacun avec ses propres caractéristiques :

- **Arbre binaire complet** : Un arbre binaire complet est un arbre où chaque niveau est complètement rempli de nœuds, à l'exception éventuellement du dernier niveau, qui peut être partiellement rempli de gauche à droite. Tous les nœuds ont soit deux fils, soit aucun fils. Les arbres binaires complets sont souvent utilisés dans la conception d'algorithmes de tri.
- **Arbre binaire parfait** : Un arbre binaire parfait est un arbre où tous les niveaux sont complètement remplis de nœuds, et tous les nœuds ont deux fils. Les arbres binaires parfaits sont souvent utilisés dans la conception d'algorithmes de hachage.

Types d'arbres binaires

- **Arbre binaire équilibré** : Un arbre binaire est dit équilibré si pour tout nœud, la hauteur de son sous-arbre gauche et de son sous-arbre droit diffèrent au plus de 1.
- **Arbre binaire de recherche** : Un arbre binaire de recherche est un arbre binaire où chaque nœud contient une clé de tri, et où la propriété d'ordre est respectée pour toutes les clés dans l'arbre.
- **Arbre binaire file d'attente de priorité** : Un arbre binaire file d'attente de priorité est un arbre binaire dans lequel chaque nœud contient une clé de priorité, et où la propriété d'ordre est respectée pour toutes les clés dans l'arbre.

Parcours des arbres binaires

Il existe plusieurs façons de parcourir un arbre binaire, chacune ayant son propre ordre de visite des nœuds. Les trois méthodes les plus courantes pour parcourir un arbre binaire sont :

- **Parcours en ordre (ou parcours infixe)** : Dans ce parcours, les nœuds sont visités dans l'ordre suivant : gauche, nœud, droite. Autrement dit, on commence par visiter le sous-arbre gauche, puis on visite le nœud courant et enfin on visite le sous-arbre droit. Ce type de parcours est souvent utilisé pour afficher les nœuds d'un arbre binaire dans l'ordre croissant.

Parcours des arbres binaires

- **Parcours préfixe (ou parcours préordre)** : Dans ce parcours, les nœuds sont visités dans l'ordre suivant : nœud, gauche, droite. Autrement dit, on commence par visiter le nœud courant, puis on visite le sous-arbre gauche et enfin on visite le sous-arbre droit. Ce type de parcours est souvent utilisé pour créer une copie de l'arbre binaire.
- **Parcours postfixe (ou parcours postordre)** : Dans ce parcours, les nœuds sont visités dans l'ordre suivant : gauche, droite, nœud. Autrement dit, on commence par visiter le sous-arbre gauche, puis on visite le sous-arbre droit et enfin on visite le nœud courant. Ce type de parcours est souvent utilisé pour évaluer des expressions arithmétiques écrites en notation polonaise inversée.

Parcours des arbres binaires

Voici un exemple d'arbre binaire et les parcours correspondants :

- **Parcours en ordre** : D, B, E, A, F, C, G
- **Parcours préfixe** : A, B, D, E, C, F, G
- **Parcours postfixe** : D, E, B, F, G, C, A

