

Activité : Comprendre les arbres en Python

Objectifs :

- Utiliser le vocabulaire de base des arbres binaires (racine, nœud, feuille, hauteur...).
- Comprendre comment construire un arbre binaire.
- Compter les nœuds pour avoir la taille de l'arbre.
- Implémenter un arbre en Python.

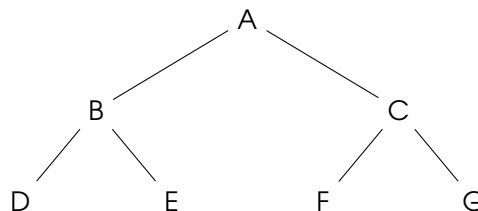
1. Rappel du vocabulaire de base

Un **arbre binaire** est une structure composée de nœuds. Chaque nœud peut avoir au plus deux enfants : un *fil gauche* et un *fil droit*.

- Le **nœud racine** est le point de départ de l'arbre.
- Une **feuille** est un nœud sans enfants.
- La **profondeur** d'un nœud c'est le nombre d'arêtes (ou de niveaux) entre la racine et ce nœud.
- La **hauteur** d'un arbre est la plus grande profondeur parmi tous ses nœuds, c'est-à-dire la distance (en nombre d'arêtes) entre la racine et la feuille la plus éloignée.

2. Construire un arbre et utiliser le vocabulaire

1. Complète le vocabulaire sur cet arbre :



Identifie : la racine, les feuilles, la profondeur du nœud C.

2. Qui sont les fils de A ?

3. (a) Représenter un arbre qui vérifie les conditions suivantes :

- Le nœud racine est M.
- M a deux enfants : E à gauche et T à droite.
- E a pour enfants A et L ; T a pour enfant R.
- A a pour enfant K.
- R a pour enfants I et P.

Indique les feuilles.

(b) L'arbre obtenu est-il unique ? Autrement dit, existe-t-il un autre arbre, différent (par exemple avec certains nœuds échangés de place), qui respecte exactement les mêmes conditions ?

(c) Combien de tels arbres vérifient ces conditions ?

- (d) Comment aurait-on pu rendre cet arbre unique ?
4. Construire par vous-même un arbre de hauteur 5 contenant au moins 5 feuilles. Combien y-a-t-il de nœuds au total ?
 5. Lorsque l'on dispose d'un arbre T , on appelle taille de T le nombre de nœuds que contient T .
 - (a) Dessiner un arbre de taille 10.
 - (b) Dessiner un arbre de hauteur 2 où chaque parent dispose exactement de deux fils. Quelle est la taille d'un tel arbre ?
 - (c) Même question pour un arbre de hauteur 3 et 4.
 - (d) Et maintenant si on dispose d'un arbre de profondeur $n \in \mathbb{N}$ dont chaque parent dispose exactement de deux enfants, conjecturer la taille d'un tel arbre.

3. Comprendre comment représenter un arbre en Python

Une liste est une structure qui permet de stocker plusieurs éléments dans un certain ordre. On peut y mettre des nombres, des chaînes de caractères, ou même d'autres listes !

Exemple :

```
nombres = [1, 2, 3, 4]
mots = ["bonjour", "salut", "hello"]
liste_de_liste = [[1,2,3], [4,5,6]]
```

Pour obtenir la taille d'une liste L on utilise la commande `len(L)`. Enfin, pour accéder au premier élément de la liste L on utilise la commande `L[0]` puis `L[1]` pour le deuxième élément *et cetera* jusqu'au dernier élément de la liste.

7. Écrire une liste en Python contenant les chiffres de 1 à 5.
8. Demander à Python d'afficher la taille de la liste précédente en utilisant la fonction `len`.
9. Écrire une liste en Python contenant d'autres listes contenant des nombres.
10. Quelle est la taille de votre liste précédente ? Comparer avec ce que vous donne Python.
11. Comment faire pour avoir le nombre d'éléments au total dans une liste de liste ?
12. Compléter la fonction Python suivante pour qu'elle détermine le nombre total d'éléments dans une liste de liste :

```
def taille_totale(L):
    taille = 0
    for l in L:
        taille = taille + _____
    return _____
```

Dans le cas où vous ne vous souvenez plus la boucle `for` appelez-moi !

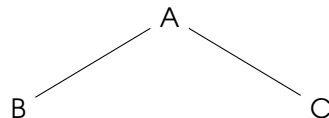
Comment utiliser des listes pour représenter un arbre ?

On représente un nœud d'un arbre binaire comme une liste contenant trois éléments :

- La valeur du nœud (une chaîne de caractères par exemple),
- Le sous-arbre gauche (encore une liste !),
- Le sous-arbre droit (une autre liste).

Exemples :

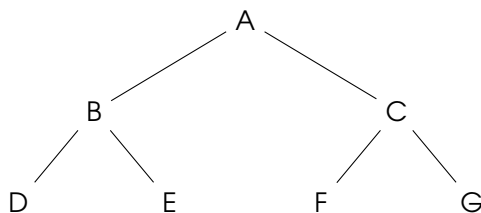
- L'arbre T suivant



est représenté en Python par :

```
T1 = ["A", ["B", [], []], ["C", [], []]]
```

- L'arbre T_2 suivant



est représenté en Python par :

```
T2 = ["A", ["B", ["D", [], []], ["E", [], []]], ["C", ["F", [], []], ["G", [], []]]]
```

13. Dessiner un arbre de hauteur 2 et représenter le en Python avec une liste de listes.
14. Comment obtenir la taille totale de l'arbre en utilisant la liste précédente ?
15. Compléter le code suivant permettant d'obtenir la taille de l'arbre :

```
def taille_arbre(T):  
    if T == []:  
        return _____  
    return 1 + taille_arbre(_____) + taille_arbre(_____)
```