Exercices - Structures hiérarchiques : arbres

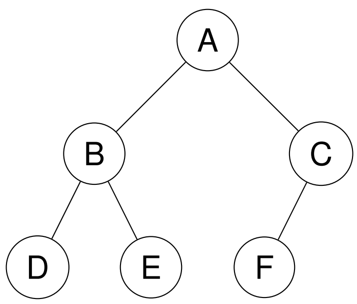
Exercice 1

1. Donner la définition d'un arbre binaire.
2. Donner la définition de la taille d'un arbre.
3. Donner la définition de la profondeur d'un noeud.
4. Donner un encadrement de la hauteur d'un arbre de taille 12.

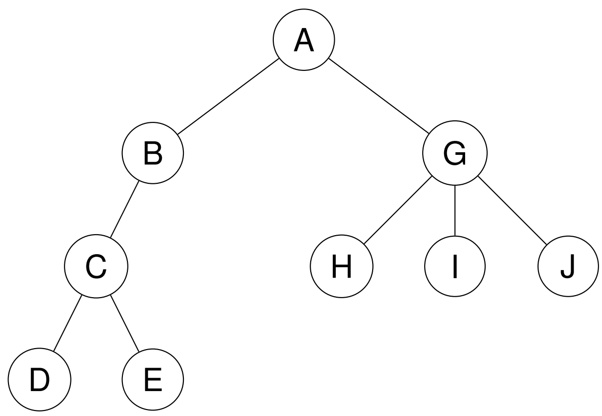
Exercice 2

Pour chacun des arbres suivants :

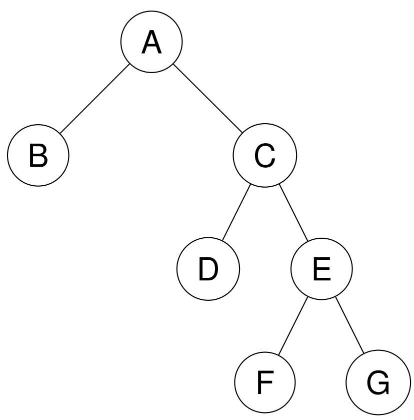
- indiquer s'il s'abit d'un arbre binaire (justifier).
- indiquer la racine de l'arbre.
- lister les feuilles de l'arbre.
- donner la taille de l'arbre.
- donner la hauteur de l'arbre.
- donner la profondeur du noeud D.
- s'il s'agit d'un arbre binaire, donner le sous-arbre gauche du noeud C.
- s'il s'agit d'un arbre binaire, donner le sous-arbre droit du noeud F.



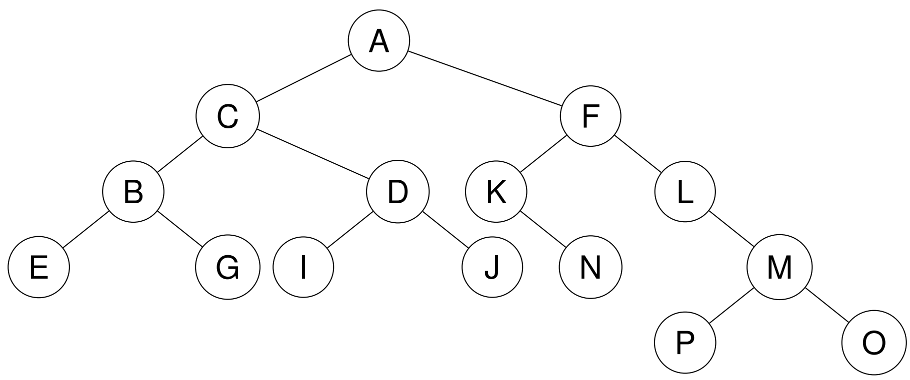
Arbre 1



Arbre 2



Arbre 3



Arbre 4

Exercice 3

Reprendre les arbres 1, 3 et 4 de l'exercice 2 et donner la liste des étiquettes selon les algorithmes de parcours :

1. largeur d'abord.
2. pronfondeur infixe.
3. profondeur prefixe
4. profondeur suffixe.

Exercice 4

On donne ci-contre le tableau caractérisant un arbre :

1. Représenter l'arbre correspondant.
2. Quelle est la hauteur de cet arbre ?
3. Quelle est la taille de cette arbre ?
4. Cet arbre est-il binaire ?
5. Cet arbre est-il parfait ?
6. Donner la liste des étiquettes de cette arbre selon le parcours en profondeur infixe.
7. Quel est le résultat de cette expression mathématique ?

Nœud	Etiquette	Nœud du SAG	Nœud du SAD
1	*	2	3
2	+	4	5
3	-	6	7
4	3		
5	/	8	9
6	8		
7	*	10	11
8	4		
9	2		
10	2		
11	3		

Exercice 5

1. Donner la définition d'un arbre binaire de recherche.
2. Dans un arbre binaire de recherche (ABR), où se trouve le plus petit élément ?
3. Quel algorithme de parcours permet d'afficher la liste des valeurs dans l'ordre croissant ?
4. On donne ci-dessous une liste aléatoire de 14 nombres entiers :
   25 ; 60 ; 35 ; 10 ; 5 ; 20 ; 65 ; 45 ; 70 ; 40 ; 50 ; 55 ; 30 ; 15
   Construire dans l'ordre de la liste l'arbre binaire de recherche associé.
5. On veut chercher l'étiquette "70". Expliquer comment procéder en partant de la racine.
6. Même question pour chercher l'étiquette "5".
7. Même question pour chercher l'étiquette "150".

Exercice 6

On donne la suite d'instructions suivantes :

```
A = CREER_ARBRE(2,CREER_ARBRE_FEUILLE(4), CREER_ARBRE_FEUILLE(3))
B = CREER_ARBRE(5,CREER_ARBRE_VIDE(), CREER_ARBRE_FEUILLE(6))
C = CREER_ARBRE(1, A, B)
```

1. Représenter le résultat de ces instructions sous la forme d'un arbre.
2. Donner l'arbre correspondant à l'instruction : `T = SAD(C)`
3. Quel est la valeur retournée par l'instruction suivante : `r = RACINE(B)`

Exercice 7

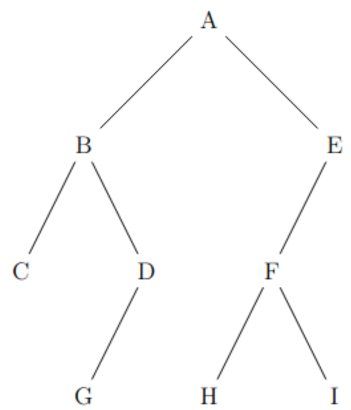
Ci-dessous une implémentation Python d'une classe `Noeud` :

```
class Noeud:
    def __init__(self, e, g=None, d=None):
        self.etiquette = e
        self.gauche = g
        self.droit = d

    def est_feuille(self):
        return not self.gauche and not self.droit

    # Une representation possible de l'arbre
    def __repr__(self):
        ch = str(self.etiquette)
        if self.gauche or self.droit:
            ch = ch + '-(' + str(self.gauche) + ',' + str(self.droit) + ')'
        return ch
```

1. Ecrire les instructions python permettant de créer les arbre 1 et 3 de l'exercice 2.
2. Ecrire une fonction python `taille`, prenant en paramètre un objet `a` de type `Noeud`, qui renvoie la taille de l'arbre `a`.
3. Ecrire une fonction python `parcours_suffixe`, prenant en paramètre un objet `a` de type `Noeud`, qui affiche la liste des étiquettes de l'arbre `a` selon le parcours en profondeur suffixe.
4. Ecrire une fonction python `recherche`, prenant en paramètre un objet `a` de type `Noeud` et un entier n, qui retourne `True` si la valeur n appartient à l'arbre `a` et `False` sinon.
   Nous faisons l'hypothèse que l'arbre `a` est un **arbre binaire de recherche**.

Exercice 8 Extrait BAC NSI

Dans cet exercice, on utilisera la convention suivante : la hauteur d'un arbre binaire ne comportant qu'un noeud est 1.



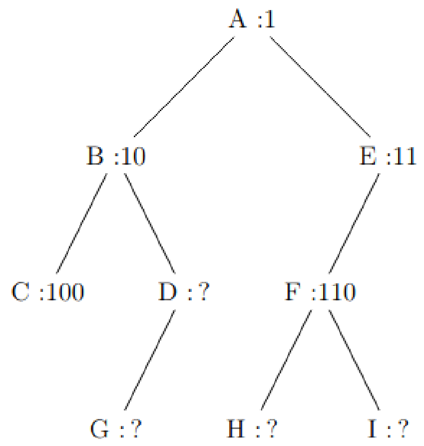
Question 1
Déterminer la taille et la hauteur de l'arbre binaire ci-contre.

Question 2
On décide de numéroter en binaire les noeuds d'un arbre binaire de la façon suivante :

- la racine correspond à 1 ;
- la numérotation pour un fils gauche s'obtient en ajoutant le chiffre 0 à droite au numéro de son père ;
- la numérotation pour un fils droit s'obtient en ajoutant le chiffre 1 à droite au numéro de son père ;

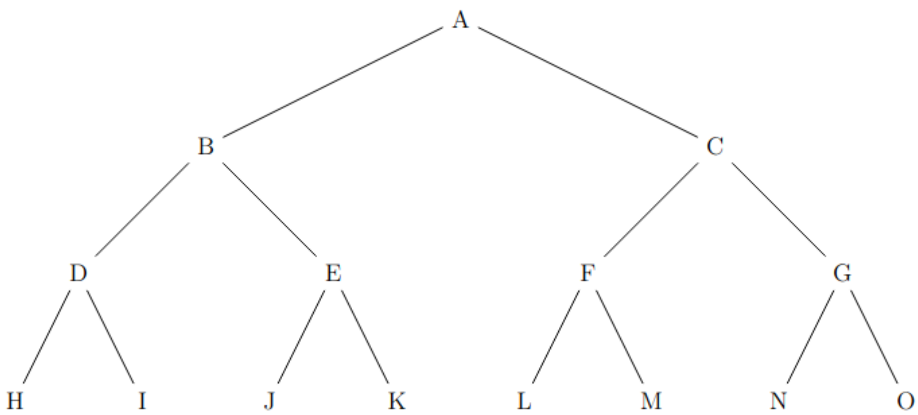
Par exemple, dans l'arbre ci-dessous, on a utilisé ce procédé pour numéroter les nœuds A, B, C, E et F .

1. Dans l'exemple précédent, quel est le numéro en binaire associé au nœud G ?
2. Quel est le nœud dont le numéro en binaire vaut 13 en décimal ?
3. En notant `h` la hauteur de l'arbre, sur combien de bits seront numérotés les noeuds les plus en bas ?
4. Justifier que pour tout arbre de hauteur h et de taille $n \geq 2$, on a : $h \leq n \leq 2^h - 1$.



Question 3

Un arbre binaire est dit complet si tous les niveaux de l'arbre sont remplis.



Arbre binaire complet

On décide de représenter un arbre binaire complet par un tableau de taille $n+1$, où n est la taille de l'arbre, de la façon suivante :

- La racine a pour indice 1 ;
- Le fils gauche du nœud d'indice i a pour indice $2 \times i$;
- Le fils droit du nœud d'indice i a pour indice $2 \times i + 1$;
- On place la taille n de l'arbre dans la case d'indice 0.

1. Déterminer le tableau qui représente l'arbre binaire complet de l'exemple précédent.
2. On considère le père du noeud d'indice i avec $i \geq 2$. Quel est son indice dans le tableau ?