

exercices-arbre-binaire.zip sur site

Télécharger exercices-arbre-binaire.zip sur le site <https://cviroulaud.github.io>

Exercice 1 : L'arbre figure 1 représente une opération arithmétique.

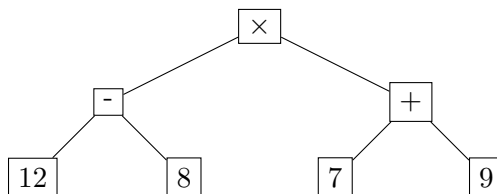


FIGURE 1 – Calcul arithmétique

1. Parcourir cet arbre en profondeur (préfixe, infixe et postfixe).
2. Donner le résultat du calcul.
3. Pour quel parcours est-il indispensable de rajouter des parenthèses ?

Exercice 2 :

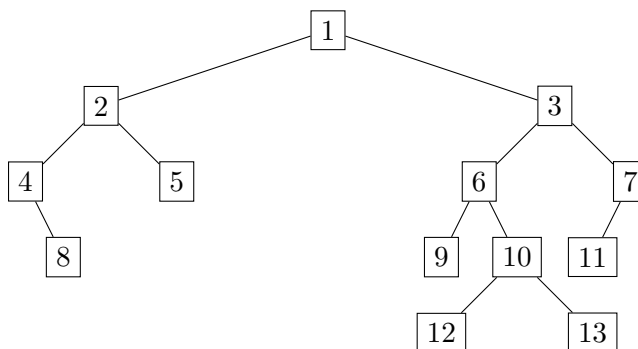


FIGURE 2 – Arbre binaire

1. Effectuer les parcours suivants dans l'arbre figure 2 :
 - en largeur,
 - préfixe,
 - infixe,
 - postfixe.
2. Quelle est la hauteur de cet arbre. On considère qu'un arbre vide à une hauteur de -1.
3. Cet arbre est-il équilibré ?
4. Cet arbre est-il complet ?

Exercice 3 : En généalogie le principe de la numérotation de Sosa-Stradonitz attribue le numéro 1 à l'individu racine (le sujet sur lequel on établit l'ascendance) puis le numéro 2 à son père et 3 à sa mère. Ainsi, chaque homme a un numéro double de celui de son enfant et donc pair et chaque femme un numéro double de celui de son enfant plus un, soit un numéro impair.

1. Le numéro 17 est-il une femme ou un homme ? Quel est les numéros de son père et sa mère ? Quel est le numéro de son enfant ?
2. Combien de trisaïeux (ancêtres de quatrième degré) possède un individu ?

3. Un généalogiste propose d'imprimer l'arbre généalogique d'un individu avec quatre degrés d'ascendance. Combien de personnes seront affichées dans cet arbre ?

L'arbre des ascendants d'Emmanuel Macron est représenté dans un liste Python (code 1).

```
1 # Pour respecter la numérotation de Sosa-Stradonitz la première cellule est  
   laissée vide  
2 macron = ["", "Emmanuel Macron", "Jean-Michel Macron", "Françoise Noguès",  
3           "André-Henri Macron", "Jacqueline Robert", "Jean Noguès",  
4           "Germaine Arribet", "Henri Eugène Macron", "Marie Adèle Bosseur",  
5           "Georges William Robert", "Suzanne Leblond", "Fabien Noguès",  
6           "Esther Mas", "Ernest Arribet", "Marie Madeleine Millet"]
```

Code 1 – Représentation en mémoire d'un arbre généalogique

4. Représenter l'arbre généalogique sous forme d'un arbre binaire.
5. Ouvrir le fichier *arbre-genealogique.py*.
6. Écrire la fonction **get_parents(tab : list, id_enfant : int) → tuple** qui renvoie le nom des parents de *id_enfant* sous forme de tuple. La fonction devra gérer les identifiants trop grands à l'aide d'une assertion.
7. Écrire la fonction **ascendant_homme(tab : list, hommes : list) → list** qui renvoie la liste des hommes de la personne dont on a réalisé l'arbre généalogique. La fonction devra parcourir l'arbre en profondeur. De plus une simple liste Python fera office de pile.
8. Écrire une version récursive de la fonction précédente.

nombre de feuilles (récursif)
insertion, suppression
binarytree
tri par tas
numérotation de Sosa-Stradonitz