

<p style="text-align: center;">info421 : Programmation fonctionnelle TD 3 : arbres binaires</p>

Responsable : Pierre Hyvernats
Laboratoire de mathématiques de l'université de Savoie
email : Pierre.Hyvernats@univ-savoie.fr
<http://lama.univ-savoie.fr/~hyvernats/>

Exercice 1 : les arbres binaires

On considère les arbres binaires avec des données sur les noeuds. Autrement dit, un arbre étiqueté par des entiers est :

- soit l'arbre vide
- soit un noeud étiqueté (par une donnée) avec deux fils : le fils gauche (un arbre) et le fils droit (un arbre).

Question 1. Donnez le type inductif correspondant.

Question 2. Écrivez la fonction `taille` de type `'a arbre -> int` qui calcule combien de noeuds possède un arbre.

Question 3. Écrivez un équivalent de la fonction `List.map` qui applique une fonction à tous les noeuds d'un arbre. Son type sera `('a -> 'b) -> 'a arbre -> 'b arbre`.

Question 4. Écrivez la fonction qui prend en argument une fonction de poids, et calcule la somme des poids des noeuds d'un arbre : `poids : ('a -> int) -> 'a arbre -> int`

Question 5. Programmez une fonction `fold` similaire à la fonction `List.fold_right` et servez vous en pour redonner une définition directe de `taille`, `applique` et `poids`.

Quel est le type de cette fonction ?

Question 6. Utilisez la fonction `fold` pour programmer la fonction qui collecte toutes les données d'un arbre dans une liste.

Question 7. Programmez une fonction qui calcule la liste des *feuilles* d'un arbre

- directement avec un `let rec`,
- en utilisant la fonction `fold`.

Exercice 2 : arbres binaires de recherche

Un arbre binaire contenant des entiers est dit "de recherche" si toutes les données à gauche d'un noeud sont plus petites que la donnée du noeud, et si toutes les données à droite d'un noeud sont plus grande que la donnée du noeud.

Question 1. Donnez le type des arbres binaires de recherche.

Question 2. Écrivez une fonction qui teste si un élément apparait dans un arbre binaire.

Quelle est la complexité de cette fonction au pire cas ?

Question 3. Écrivez une fonction qui insère un élément dans un arbre binaire de recherche.

Question 4. Écrivez une fonction qui supprime un élément dans un arbre binaire de recherche.

Question 5. Écrivez une fonction qui vérifie si un arbre binaire est bien un arbre binaire *de recherche*.