

Variante de `filter` pour les ABR

Consigne Comme dans votre recueil d'exercice, on utilisera le type `ArbreBinRecherche` pour représenter les arbres binaires de recherche dont les éléments sont des nombres. On ne considérera que de tels ABR.

Votre recueil d'exercice contient plusieurs exemples d'arbres binaires de recherche que vous pourrez utiliser pour vos tests. Notamment dans l'exercice 11.

Question 1 Écrivez une *spécification* et une *définition* pour la fonction `abr-max` qui retourne le plus grand élément d'un ABR.

Question 2 Pour supprimer un élément `x` d'un ABR, on utilisera l'algorithme suivant:

- Si `x` est à une feuille `f` (c'est-à-dire que ses deux sous-arbres de `f` sont vides), on peut remplacer `f` par l'arbre vide.
- Si `x` est dans un nœud `n` qui a exactement un sous-arbre vide, on peut remplacer `n` par son sous-arbre non-vide
- Si `x` est dans un nœud `n` qui a ses deux sous-arbres non-vides. Soit `y`, l'élément maximum de son sous-arbre gauche. On remplace `n` par l'arbre d'étiquette `y`, de même sous-arbre droit que `n`, et dont le sous-arbre gauche est le sous-arbre gauche de `n` dans lequel on a préalablement supprimé `y`

Écrivez la *spécification* et la *définition* d'une fonction `abr-suppr` qui supprime un élément d'un ABR en employant cette méthode. Si `x` n'est pas dans `A`, `abr-suppr x A` doit retourner `A`. Rappelez-vous qu'un ABR contient un élément au plus une fois.

Question 3 En utilisant la fonction `abr-suppr`, écrivez la *spécification* et la *définition* d'une fonction `abr-filter-1` qui, étant donné un ABR `A` et un prédicat `p?` sur les nombres, retire de `A` tous les éléments qui ne satisfont pas `?p`.

Question 4 Remarquez comme le plus grand élément d'un ABR a toujours un sous-arbre droit vide. Par conséquent, il est facile de le supprimer. On peut tirer avantage de ce fait en supprimant le plus grand élément en même temps qu'on le cherche.

Écrivez la *spécification* et la *définition* d'une fonction `abr-suppr-max` qui, étant donné un ABR `A`, renvoie un couple constitué du plus grand élément de `A` et de `A` privé de son plus grand élément. La fonction `abr-suppr-max` ne doit faire qu'un seul parcours de `A`.

Question 5 Utilisez la fonction `abr-suppr-max` pour écrire la *définition* d'une fonction `abr-filter` de même spécification de `abr-filter-1`, mais qui n'utilise pas `abr-max` ou `abr-suppr`.