TD19 : Arbres rouge-noir et autres

Exercice 1

1. Donner l'arbre rouge-noir obtenu en insérant une à une et dans l'ordre les valeurs entières 1 à 11.
2. Donner l'arbre rouge-noir obtenu en insérant une à une et dans l'ordre les valeurs entières 11 à 1.

Exercice 2 Trouver une suite de valeurs telle que la hauteur de l'arbre rouge-noir obtenu en les insérant une à une dans l'ordre dans un arbre initialement vide est supérieure à la hauteur de l'arbre binaire de recherche obtenu en insérant cette même suite de valeurs dans l'ordre dans un arbre initialement vide.

Exercice 3 Montrer que dans tout arbre binaire de recherche de taille n, il existe exactement $n-1$ rotations possibles.

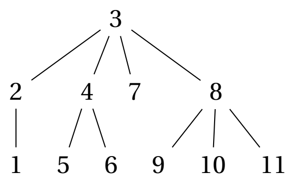
Exercice 4 Soit un entier n et deux arbres binaires de recherche A et B de taille n contenant les mêmes valeurs (qu'on suppose deux à deux distinctes pour simplifier).

1. Montrer qu'il est possible d'obtenir B à partir de A en appliquant une suite de rotations gauches et droites.
2. Montrer qu'il est possible de construire une telle suite de rotations de longueur $\mathcal{O}(n)$.
3. Montrer qu'il n'est pas toujours possible d'obtenir B à partir de A en appliquant uniquement une suite de rotations droites (sans aucune rotation gauche).
4. Montrer que, s'il est possible d'obtenir B à partir de A en appliquant uniquement une suite de rotations droites, alors il existe une telle suite de longueur $\mathcal{O}(n^2)$.

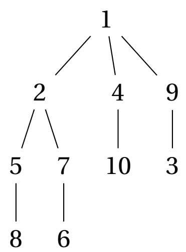
Exercice 5 (adaptation de l'exercice 18 du livre *"Les clefs pour l'info"*)
Soit un arbre T de taille n dont les nœud sont étiquetés par les entiers $\{1,\ldots,n\}$ (on identifie un nœud à son étiquette). Le *codage de Prüfer* de T est la liste L définie comme résultat de l'algorithme suivant :

$L \leftarrow$ liste vide
tant que T contient au moins 2 sommets **faire**
    $x \leftarrow$ la feuille de plus petit numéro de T
    $y \leftarrow$ père(x)
    ajouter y à L
    supprimer le nœud x de T
fin tant que
renvoyer L

Par exemple de codage de Prüfer de l'arbre suivant est `[2,3,4,4,3,3,8,8,8]` :



1. Donner le codage de Prüfer de l'arbre suivant :



2. Donner un arbre associé au codage de Prüfer `[1,1,1,1,1,1,1]`
3. On appelle *degré* d'un nœud le nombre de nœuds auxquels il est relié (c'est donc son arité +1 pour les nœuds qui ne sont pas la racine, et son arité pour la racine). Montrer qu'un nœud de l'arbre est de degré k si et seulement s'il apparaît $k-1$ fois dans le codage de Prüfer.
4. Proposer une méthode pour reconstruire l'arbre à partir de son codage de Prüfer et de son tableau de degrés.