TD Algo2 – session 4 – Arbres rouge-noir

12 mars 2025

Objectifs d'apprentissage :
- manipuler et étudier l'opération de rotation et concevoir des algorithmes relatifs ;
- manipuler la structure d'arbre rouge-noir et étudier les implications des propriétés de ce type d'arbres ;

> Les 7 premiers exercices sont essentiels. Plus d'exercices sur `https://algs4.cs.princeton.edu/33balanced/`.

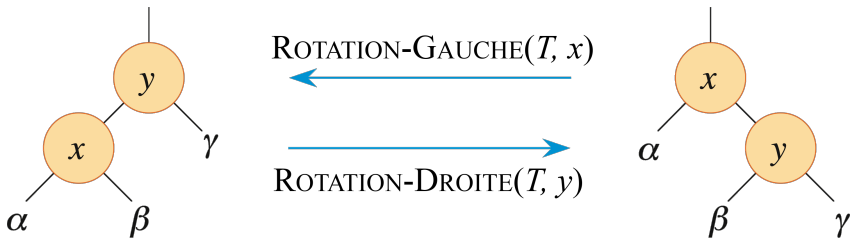
Exercice 1 : Insertion

On considère ce qui se passe lorsqu'un arbre rouge-noir est construit en insérant des clés dans l'ordre croissant et on s'intéresse au nombre de rotations réalisées. Déterminer, en l'argumentant, un nombre maximal de rotations réalisées pour l'insertion de n clés. La borne n'a pas besoin d'être précise à un facteur constant près.

> On peut faire un exemple avec quelques valeurs et on constate qu'il n'y a jamais plus d'une rotation par insertion. Le nombre de rotations semble être en $O(n)$.
>
> On peut argumenter en observant que chaque insertion entraîne au maximum deux rotations (cas 2 et 3). Il ne peut donc pas y avoir plus de $2n$ rotations.
>
> On peut affiner en constatant que le cas 2 n'apparaît jamais car chaque nouveau nœud est inséré sur le bord droit. Mais il faudrait un argument plus fort pour le prouver et affirmer qu'il ne peut pas y avoir plus de n rotations.

Exercice 2 : Rotation droite

Écrire le pseudo code de ROTATION-DROITE.



> Les 4 premières instructions sont une sécurité pour ne perdre la référence d'aucun nœud en cours d'exécution (inutile ici).

```
ROTATION-DROITE(T, y)
p ← y.parent
α ← x.gauche
β ← x.droite
γ ← y.droite
x ← y.gauche
y.gauche ← x.droite
si x.droite ≠ NIL alors
   x.droite.parent ← y
x.parent ← y.parent
si y.parent = NIL alors
   T.racine ← x
sinon si y = y.parent.droite alors
   y.parent.droite ← x
sinon
   y.parent.gauche ← x
x.droite ← y
y.parent ← x
```

Exercice 3 : Rotations possibles

Argumenter que, dans tout arbre binaire de recherche à n nœuds, il existe exactement $n-1$ rotations possibles.

Une rotation gauche peut avoir lieu si un nœud et son enfant de droite existe (et inversement pour une rotation droite).

Pour chaque nœud qui est un enfant de droite, il est donc possible de réaliser une rotation gauche sur son parent (et inversement pour un enfant de gauche).

Comme tous les nœuds sont soit en enfant de droite, soit une enfant de gauche, à l'exception de la racine, il y a donc un total de $n-1$ rotations possibles.

Exercice 4 : Transformation

On cherche un algorithme pour transformer tout arbre binaire de recherche à n nœuds en n'importe quel autre arbre binaire de recherche à n nœuds. Quelle est la complexité en temps de l'algorithme ?

(Conseil : commencer par identifier comment réaliser $n-1$ rotations droites pour transformer l'arbre en une chaîne orientée vers la droite.)

Réaliser une rotation droite sur la racine si elle a un enfant gauche. Sinon, répéter l'algorithme récursivement en considérant le sous-arbre enraciné en son enfant de droite.

```
TRANSFORMER-CHAÎNE(T)
x ← T.racine
tant que x ≠ NIL faire
   si x.gauche ≠ NIL alors
      Afficher("Rotation-Droite", x)
      Rotation-Droite(T, x)
   sinon
      x ← x.droite
```

On applique cet algorithme aux deux arbres binaires de recherche. Pour passer de l'un a l'autre, on réalise d'abord les rotations droites sur le premier pour obtenir une chaîne puis on applique les rotations inverses pour obtenir le second.

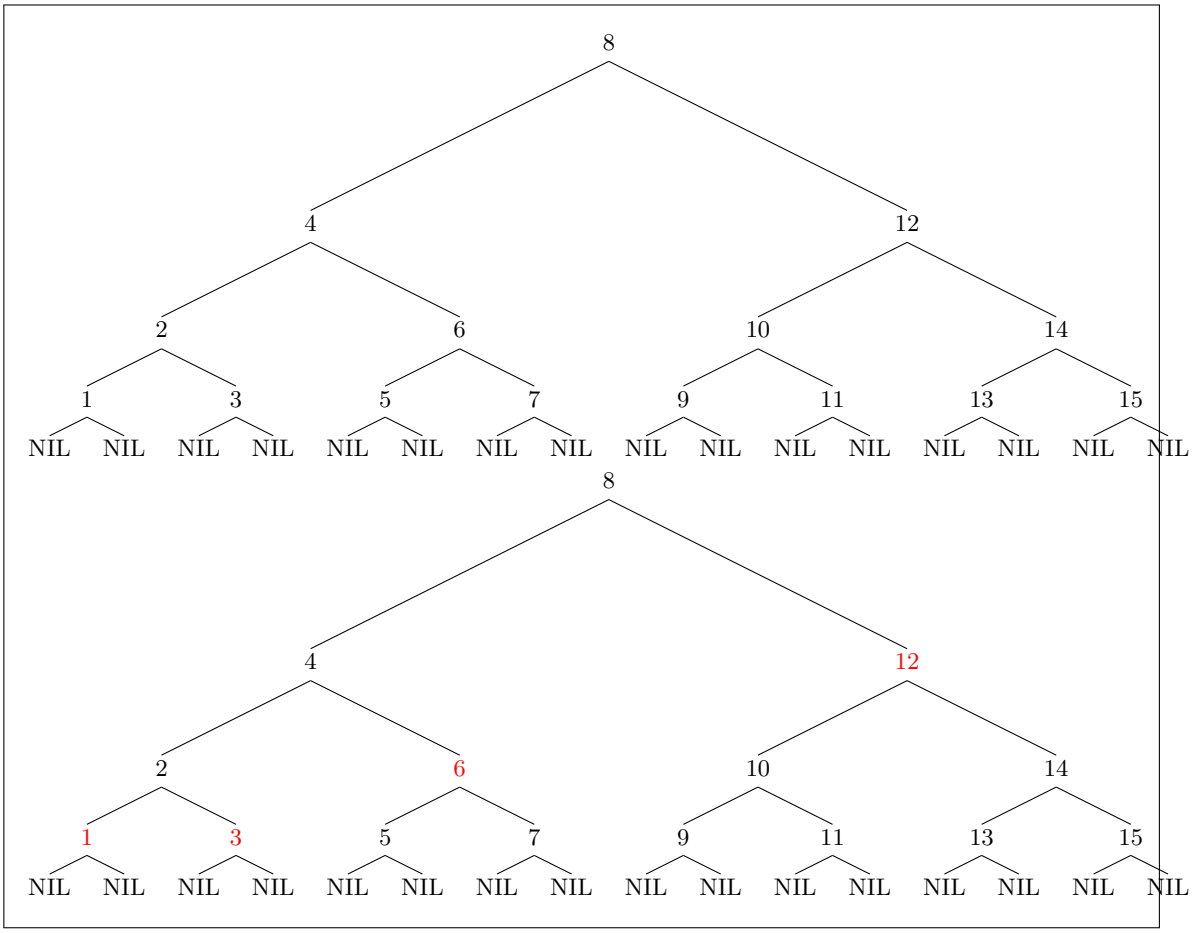
```
TRANSFORMER-CHAÎNE-INVERSE(T)
x ← T.racine
tant que x ≠ NIL faire
  si x.gauche ≠ NIL alors
    Afficher("Rotation-Gauche", x.gauche)
    Rotation-Droite(T, x)
  sinon
    x ← x.droite
```

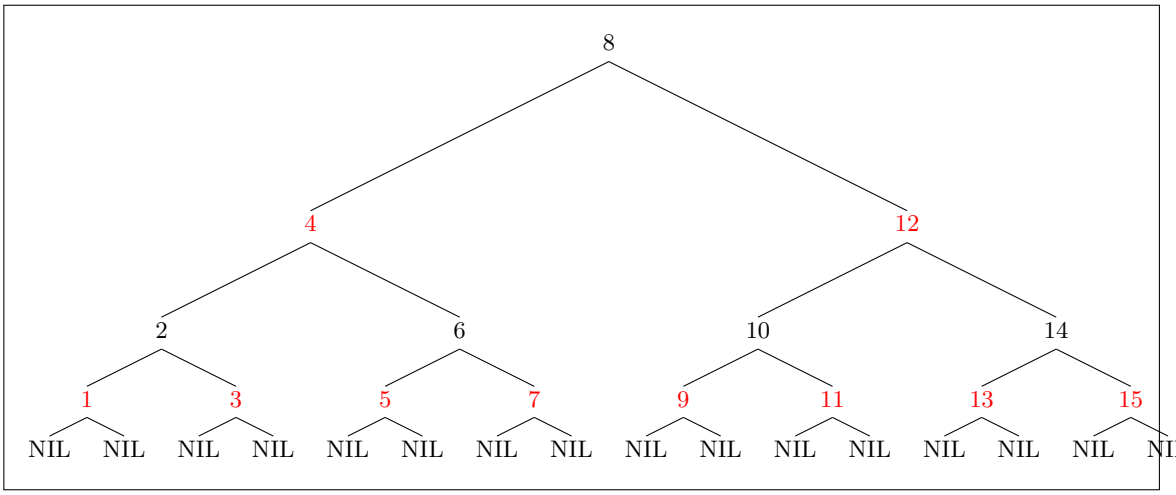
On lance le premier algorithme sur le premier arbre et le second sur l'autre, on inverse la seconde séquence de rotations et on concatène l'ensemble des rotations.

La complexité dépend du nombre de rotations qu'il faut réaliser pour passer d'un arbre quelconque à une chaîne. Dans tout arbre, on peut considérer que l'on a déjà un début de chaîne avec la racine et tous les nœuds qui suivent en ne considérant que les arêtes de droite (chacun de ces nœuds pouvant avoir des enfants à gauche). Dès qu'on réalise une rotations à droite, on ajoute un nœud sur cette chaîne. On ne peut donc faire au maximum que $n-1$ rotations. La complexité en temps est donc $O(n)$.

Exercice 5 : Coloriage

Dessiner l'arbre binaire de recherche complet de hauteur 3 contenant les clés $1, 2, \ldots, 15$. Ajouter les feuilles NIL et colorier les nœuds de trois manières différentes, de telle façon que les hauteurs noires des arbres rouge-noir résultants soient 2, 3 et 4.





Exercice 6 : Hauteur

Argumenter que le chemin simple le plus long reliant un nœud x d'un arbre rouge-noir à une feuille a une longueur qui est au plus égale à deux fois celle du plus court chemin simple reliant le nœud x à une feuille.

> Dans un cas extrême, tous les nœuds entre x et une feuille sont noirs. Soit $hn(x)$ la hauteur noire de x. Il y en a donc au moins $hn(x)$ nœuds sur un chemin de x à une feuille (x non compris). À l'inverse, s'il y a un maximum de nœuds rouges entre x et une autre feuille, comme il n'y a jamais deux nœuds rouges successifs, il ne peut y en avoir que $2hn(x)$ au maximum.

Exercice 7 : Extrêmes

Quel est le plus grand nombre possible de nœuds internes d'un arbre rouge-noir de hauteur noire k ? Quel est le plus petit nombre possible ?

> Avec que des nœuds noirs, si la hauteur noire est 1, il y a au moins 1 nœud. Si c'est 2, il y en a au moins 2. Si c'est 3, il y en a au moins 4. Ça en fait donc au moins 2^{k-1}.
>
> Avec un maximum de nœuds rouges, si la hauteur noire est 1, il y en a au plus 3. Si c'est 2, il y en a au plus 15. Ça en fait donc au plus $2^{2k} - 1$.

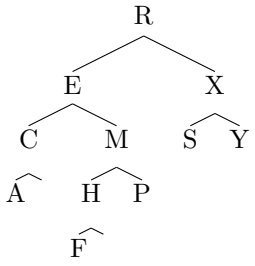
Exercice 8 : Nœud rouge

Soit un arbre rouge-noir formé par l'insertion de n nœuds via RN-Insérer. Argumenter que, si $n > 1$, l'arbre a au moins un nœud rouge.

> À chaque insertion, un nœud est initialement rouge. Il n'y a que les autres nœuds qui peuvent changer de couleur et donc il y a toujours au moins un nœud rouge, le dernier inséré. Pour $n = 1$, il n'y a que la racine et c'est le seul cas où sa couleur devient rouge avec la dernière instruction.

Exercice 9 : Rotation

Supposons que l'on réalise une rotation à gauche pour le nœud contenant E dans l'arbre suivant. Quel est le résultat d'un parcours en largeur sur l'arbre résultant ?



R M X E P S Y C H A F