

Arbre cousu

Qu'est-ce que l'arbre binaire cousu ?¹

Un arbre binaire est cousu en faisant pointer tous les pointeurs enfants droits qui seraient normalement nuls vers le successeur inorder du nœud (s'il existe), et tous les pointeurs enfants gauche qui seraient normalement null pointent vers le prédécesseur inorder du nœud.

- Nous avons les pointeurs référençant le nœud suivant dans une traversée en ordre; appelé threads
- Nous avons besoin de savoir si un pointeur est un lien réel ou un thread, pour cela nous gardons donc un booléen pour chaque pointeur

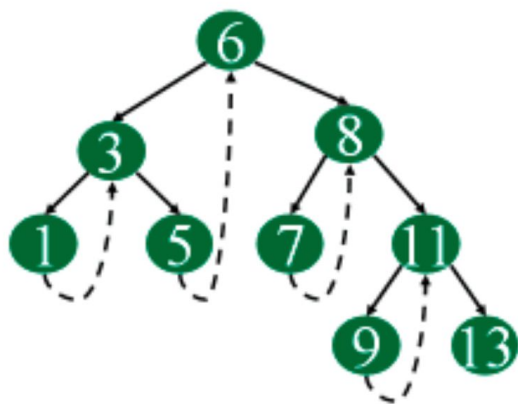
Pourquoi avons-nous besoin d'un arbre binaire fileté?

- Les arbres binaires ont beaucoup d'espace gaspillé: les nœuds feuilles ont chacun 2 pointeurs nuls. Nous pouvons utiliser ces pointeurs pour nous aider dans les parcours en ordre.
- L'arbre binaire cousu accélère le parcours de l'arbre car nous n'avons pas besoin de pile ou de récursivité pour la traversée.

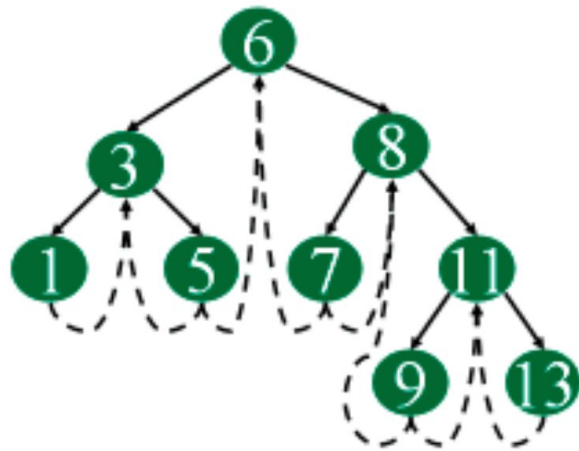
Types d'arbres binaires filetés:

- **Uniquement cousu** : chaque nœud est cousu vers le prédécesseur ou le successeur dans l'ordre (gauche ou droite). Ce qui signifie que tous les pointeurs nuls droits pointeront vers le successeur inorder OU tous les pointeurs nuls gauches pointeront vers le prédécesseur inorder.
- **Doublement cousu** : chaque nœud est cousu à la fois vers le prédécesseur et le successeur dans l'ordre (gauche et droite). Ce qui signifie que tous les pointeurs nuls droits pointeront vers le successeur inorder ET tous les pointeurs nuls gauches pointeront vers le prédécesseur inorder.

¹ <https://algorithms.tutorialhorizon.com/introduction-to-threaded-binary-tree/>



Arbre simplement cousu



Arbre doublement cousu

Tâches à Faire

En vous basant sur vos propres recherches et sur les ressources du cours, proposer un programmes avec les fonctions suivantes:

- insertion dans un arbre cousu,
- suppression dans un arbre cousu,
- recherche dans un arbre cousu,
- transformation d'un arbre binaire de recherche (ABR) en un arbre cousu,
- menu de choix pour effectuer les différentes opérations dans un arbre cousu.

Afin de mieux tester l'utilité des arbres cousus,

1. créer un ABR;
2. faire une fonction itérant sur l'insertion successive de 1000 à 10000 valeurs aléatoires dans l'ABR, avec une intervalle de 1000, c'est-à-dire, vous testez d'abord sur 1000, ensuite 2000, ensuite 3000, ..., jusqu'à 10000 (remarque: vous pouvez aller aussi au delà, rien ne vous empêche de tester avec 100000 noeuds);
3. pour chaque test à la question précédente, appeler une fonction de recherche dans cet ABR, et estimer la durée de calcul;
4. comparer les durées en temps de calcul les différentes cardinalités (1000, 2000, 3000, 4000, 5000, ..., 10000);
5. Qu'observez-vous ?
6. Refaire les étapes 2 à 4 cette fois ci avec un arbre cousu.
7. Qu'observez-vous?
8. Comparer les durées obtenues avec les ABR et celles obtenues avec les arbres cousus.
9. Quelles conclusions tirez-vous de ces étapes et comparaisons ?