

Arbres, forêts et... botanique !

Gilles Simonin

14 octobre 2021

Résumé

Durant cette séance de 2h, vous allez travailler deux aspects bien distincts de l'utilisation du concept d'arbre en informatique. Dans un premier temps, nous utiliserons ce concept afin de représenter de manière efficace la notion d'ensemble d'éléments ordonnés. Dans un second temps, nous vous demanderons de mettre en œuvre un algorithme classique de l'état de l'art pour le problème d'arbre recouvrant de point minimum.

1 Arbre binaire et structures de données

Le *tas binaire* est une des structures de données les plus classiques en informatique permettant de trier et stocker un ensemble d'éléments de manière efficace. Cette structure repose sur la classification des éléments à ordonner dans un *arbre binaire* avec la propriété que le dit arbre soit *complet* et que l'élément stocké à la racine de l'arbre soit plus petit ou égal (dans le cas *tas binaire minimum*) à tous les éléments stockés dans les sommets fils de la racine.

Nous vous proposons dans la première partie de ce TD/TP de mettre en œuvre cette structure.

Question 1 *Proposez une implémentation de la structure d'arbre binaire introduite dans le cours.*

Question 2 *Proposez une méthode d'insertion d'un nouvel élément dans le tas binaire assurant que celui-ci soit correctement positionné. Pour cela, vous opérerez de la manière suivante :*

- Ajouter l'élément dans la première feuille disponible de l'arbre ;
- Remonter cet élément jusqu'à rencontrer un parent qui soit plus petit ou égal.

Question 3 *Discutez de la complexité de l'opération d'insertion.*

Question 4 *Proposez une méthode de suppression du plus petit élément dans le tas binaire. Pour cela, vous opérerez de la manière suivante :*

- Permuter la racine de l'arbre avec la dernière feuille utilisée de l'arbre ;
- Descendre l'élément contenu dans la racine jusqu'à ce que tous les éléments contenus dans les fils du sommet courant soient plus grands.

2 Mise en œuvre - Contrôle continu 2 - Travail en Duo

Continuez avec votre binôme du travail en contrôle continu partie 1. Vous rendrez votre travail final (parties 1 et 2) avant le 30 novembre au soir dans un dépôt sur Moodle.

Nous vous proposons ici de mettre en œuvre l'algorithme de **PRIM** introduit rapidement en cours.

Question 5 *Implémentez une première version naïve du tas binaire avec des valeurs simples aux noeuds avec la classe `BinaryHeap`, en veillant à bien décomposer votre code afin de pouvoir modifier aisément chaque étape. Quel est la complexité au pire cas de votre proposition ? Développez ensuite une version améliorée dans la classe `BinaryHeapEdge`, où chaque noeud contient une arête. Quelle est la nouvelle complexité ?*

Question 6 *Comment utiliseriez-vous la structure de tas binaire dans votre mise en œuvre de l'algorithme PRIM ? Quel opération faut-il ajouter dans la structure de tas binaire ? Quel gain en terme de complexité par rapport à votre algorithme ?*