

Centre Universitaire de Mila
Institut de mathématiques et informatique
Département de l'informatique

Master 1 I2A

Année : 2025/2026

Module : Algorithmique avancée et complexité

TD 2

Exercice 1 : Arbre binaire de recherche

On veut implémenter un arbre binaire de recherche (ABR) en C++ pour stocker des nombres entiers :

- 1) Proposer la ou les structure(s) de données nécessaire (s) pour implémenter cet arbre ;
- 2) Ecrire la fonction qui crée un nouveau nœud ;
- 3) Ecrire une fonction qui cherche le minimum dans cet arbre ;
- 4) Ecrire une fonction qui retourne le successeur d'un nœud dans cet arbre ;
- 5) Ecrire une fonction qui supprime un entier de cet arbre ;
- 6) Ecrire une fonction qui compte le nombre de feuilles dans cet arbre ;

Exercice 2 : TAS

- 1) Ecrire un algorithme qui permet de vérifier si un arbre binaire complet implémenté avec un tableau est un TAS ou non ;

En considérant un TAS max d'entiers T , soient les fonctions :

insertion(T , *taille*, e) : qui insère un entier e dans T .

supprimer(T , *taille*) : qui supprime la racine du T .

max(T) qui retourne le plus grand entier dans le TAS T .

- 2) En utilisant ces fonctions, écrire un algorithme qui trie un tableau Tb de n entiers avec T (supposant que la taille de stockage de T est égale à $n \times 2$).