

Numérique et science informatique
Classe de Terminale

Lycée hoche

année scolaire 2025-2026

Contents

1	Structure d'un ABR	2
1.1	Attributs des noeuds	2
1.2	Propriétés des clés	2
2	Affichage et recherche de clés dans un ABR	2
2.1	Affichage des clés par ordre croissant	2
2.2	Recherche de la clé maximale	3
2.3	Recherche d'une clé particulière	4
2.4	Recherche du successeur d'un noeud	4
3	Ajout d'une clé dans un ABR	6
3.1	Méthode	6
3.2	Exemple	6

1 Structure d'un ABR

On s'intéresse à une classe particulière d'arbres binaires : les arbres binaires de recherche appelés également les ABR. Ceux-ci définissent des structures de données qui ont pour structure logique un arbre binaire et qui peuvent supporter des opérations courantes sur des ensembles dynamiques ; par exemple : rechercher, minimum, maximum, prédécesseur, successeur, ajouter, supprimer, etc. Ces arbres sont fondamentaux dans beaucoup de domaines : gestion des fichiers sur un disque dur, etc.

1.1 Attributs des noeuds

Un ABR étant un arbre binaire, on utilise pour ses nœuds les mêmes attributs que ceux vus dans la leçon « Généralités sur les arbres binaires » :

- $n.clé$ désigne la « valeur » contenue dans n ;
- $n.père$ désigne le père de n ;
- $n.gauche$ désigne le fils gauche de n ;
- $n.droite$ désigne le fils droit de n . On donne la valeur particulière **null** à un attribut lorsque celui-ci est vide.

1.2 Propriétés des clés

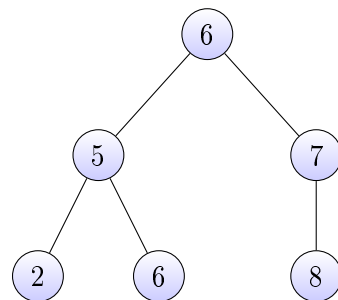
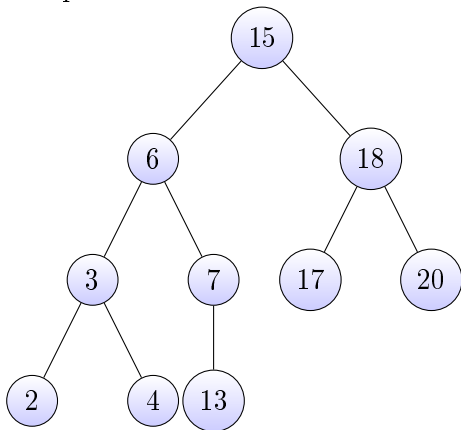
Contrairement à un arbre binaire général, les clés d'un ABR doivent être comparables entre elles. Plus précisément, pour tout nœud et pour tout descendant de , alors :

- si y est un nœud du sous-arbre gauche de x , alors $y.clé \leq x.clé$;
- si y est un nœud du sous-arbre droit de x , alors $y.clé \geq x.clé$

Remarque:

Dans la suite, nous ne considérons que des clés numériques ou chaînes de caractères.

Exemples:



(a) avec des clés numériques distinctes

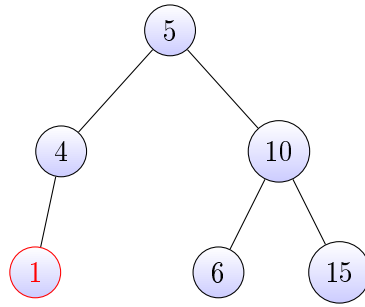
(b) avec des clés numériques non distinctes

2 Affichage et recherche de clés dans un ABR

2.1 Affichage des clés par ordre croissant

Exercice 1(Affichage des clés dans l'ordre croissant)

On considère l'ABR ci-dessous.



- Donner l’affichage des clés que l’on doit obtenir.
- Avec quel type de parcours obtient-on un tel affichage ?

Théorème 1

Un parcours en profondeur d’un ABR avec traitement infixé renvoie les clés de l’ABR par ordre croissant.

2.2 Recherche de la clé maximale

Exercice 2

1. Quelle est la clé maximale de l’ABR de l’exercice 1 ? Où est-elle située ?
2. De façon générale, où est située la clé maximale dans un ABR et comment doit-on la chercher ?

Exercice 3

Compléter l’algorithme suivant pour qu’il renvoie la clé maximale de l’ABR B :

Algorithme 1 : recherche de la clé maximale

```

fonction ARBRE_Maximum(T)
.....Compléter....
....plusieurs lignes....
fin fonction
  
```

Quelle est la complexité de cet algorithme dans le pire des cas?

Rédiger votre réponse ici:

.....

.....

.....

.....

.....

.....

2.3 Recherche d'une clé particulière

Pour rechercher une clé particulière dans un ABR, la méthode est extrêmement simple : il suffit de parcourir l'arbre depuis la racine jusqu'à la clé (ou une feuille si la clé n'existe pas) en branchant si nécessaire à chaque nœud en fonction de la valeur de la clé :

- Dans le sous-arbre gauche si $clé < noeud.clé$;
- dans le sous-arbre droit si $clé > noeud.clé$.

Exercice 4

Ecrire en Pseudo-code , une fonction itérative *abr_Rechercher*(*B, val*) qui prend pour paramètres un arbre *B* et une valeur *val* qui traduit l'algorithme décrit ci-dessus.

Algorithme 2 : Rechercher une valeur z d'un nœud dans un ABR

```

fonction ARB-Rechercher(T, val) ;
  r ← T.racine;
  .... Compléter
  .... plusieurs lignes..
fin fonction ;

```

Quelle est la complexité de cet algorithme. Rédiger une justification ci-dessous:

.....

.....

.....

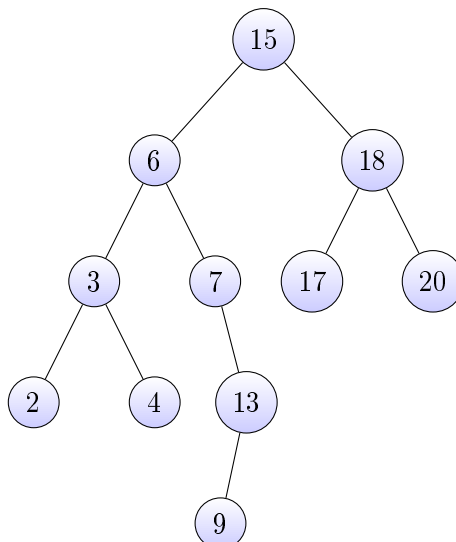
.....

2.4 Recherche du successeur d'un nœud

On souhaite déterminer le successeur d'un nœud dans l'ordre déterminé par un parcours infixé de l'arbre. Pour rendre l'algorithme très efficace, il faut réaliser cette recherche sans effectuer de comparaison entre les clés.

Exercice 5

- Quel est le successeur de 15 ? de 4 ? de 18 ? de 2 ?
- Pour la structure d'ABR de l'arbre, que représente chacun de ces successeurs vis-à-vis du nœud initial ?



Méthode:

En utilisant la structure d'un ABR, on déduit la méthode suivante pour déterminer le successeur d'un nœud x :

- si le sous-arbre droit de x est non vide, alors le successeur de x est située dans ce sous-arbre droit ;
- si le sous-arbre droit de x est vide et que x a un successeur , alors le successeur de x est de x dont est aussi

(Attention, chaque nœud est un ancêtre et un descendant de lui-même).

Exercice 5

Compléter l'algorithme suivant:

```

fonction Successeur (n)
Si .....
    renvoyer .....
Fin Si
m = .....
Tant que .....
    n = m
    .....
fin Tantque
Renvoyer m
fin fonction
    
```

Exercice 6

Ecrire une fonction Predecesseur(n) qui détermine le prédecesseur de n dans l'ordre déterminé par un parcours infixé de l'arbre.

On a le théorème suivant , qu'on déduit des algorithmes précédents:

Théorème 2

Pour un ABR, les fonctions : RECHERCHER, MINIMUM, MAXIMUM, SUCCESEUR et PRE-DECESEUR sont exécutées en temps $O(h)$, où h est la hauteur de l'ABR.

3 Ajout d'une clé dans un ABR

L'ajout d'une clé dans un ABR est une opération délicate. En effet, il faut modifier l'ensemble dynamique représenté par l'ABR tout en gardant une structure d'ABR. Il existe principalement deux méthodes d'insertion :

- l'insertion aux feuilles : on modifie dans ce cas la hauteur de l'ABR ;
- l'insertion à la racine : on modifie dans ce cas la structure même de l'ABR. L'insertion aux feuilles est bien sûr la méthode d'insertion la plus simple et c'est uniquement celle-ci que nous traitons ci-dessous.

3.1 Méthode

Pour insérer un nœud aux feuilles d'un ABR, il suffit de partir de la racine de l'ABR, de parcourir un chemin descendant jusqu'à un attribut null en testant la valeur de la clé du nœud à insérer avec la valeur de la clé de chaque nœud rencontré sur le chemin, puis de remplacer l'attribut null par le nœud voulu. Ce nœud devient alors une feuille de l'ABR.

3.2 Exemple

Dans l'exemple ci-dessous, la figure de gauche montre l'ABR initial et la figure de droite le nouvel ABR une fois que le nœud 14 a été inséré. Les nœuds en gris clair indiquent le chemin descendant de la racine vers la position où l'élément doit être inséré et la ligne en pointillés indique le lien qui a été ajouté à l'ABR pour insérer l'élément 14.

