

Midterm Algo 2027

Exercice 1 (Haches et graphes... - 6 points)

1. Donnez une méthode de hachage direct.
2. Quel problème apparaît avec le hachage coalescent?
3. Qu'est-ce qu'une collision primaire?
4. Qu'est-ce que l'ordre d'un graphe non-orienté?
5. Un sommet isolé est:

Soit le graphe $G=(S,A)$ orienté tel que : $S=1,2,3,4,5,6,7,8,9$ et

$A=(1,2),(1,6),(2,3),(2,5),(3,1),(3,4),(3,5),(4,5),(4,8),(6,2),(6,5),(7,5),(7,6),(7,8),(8,5),(8,9),(9,4),(9,7)$

6. Remplir le tableau des demi-degrés extérieurs des sommets G .
7. Le graphe est-il fortement connexe?
8. Si non, combien possède-t-il de composantes fortement connexes?
9. S'ils existent, les sommets de G de degré différent de 4 sont:
10. S'ils existent, les sommets de G de demi-degré intérieur égal à 0 sont:

Exercice 2 (Level from x - 5 points)

Insérer figure 1 ici
Écrire la fonction `keys_after(T:tree,x:int)` qui retourne la liste des clés dans l'arbre général T (implémentation "classique") se trouvant après la **première** valeur x trouvée sur le même niveau que x .

Exercice 3 (Average subtrees - 6 points)

Dans un arbre général, on cherche à savoir si un noeud n'a aucun sous-arbre dont la moyenne des clés est strictement supérieure à la clé de ce noeud.

Par exemple, l'arbre de la figure 2 ne respecte pas la propriété: la moyenne des clés du premier sous-arbre du noeud contenant 4 est supérieure à 4.

Insérer figure 2 et 3

Par contre l'arbre de la figure 3 respecte bien la propriété :

- Le noeud contenant 15:
 - la moyenne des clés du premier sous-arbre est 2
 - la moyenne des clés du deuxième sous-arbre est $5 = (8+11+2+5+0+4)/6$
- Noeud contenant 8: la moyenne des clés du premier sous-arbre est 5, inférieure à 8

- Tous les autres sous-arbres sont réduits à une feuille dont la clé est inférieure à celle du père.

Écrire la fonction `average_subtrees(B:TreeAsBin)` qui vérifie si l'arbre B en implémentation premier fils-frère droit respecte la propriété.

Exercice 4 (B-arbre: insertions et suppressions - 3 points)

Pour chaque question, utiliser le principe "à la descente" (principe de précaution) vu en td (hors bonus).

1. Dessiner l'arbre après insertions successives des valeurs 21, 42 et 8 dans l'arbre de la figure 4.
Insérer figure 4
2. Dessiner l'arbre après suppression de la valeur 80 dans l'arbre de la figure 5.
Insérer figure 5

Annexes

- Arbres généraux
 - Implémentation classique
 - * T : classe Tree
 - * T.key
 - * T.children: listes des fils ([] pour les feuilles)
 - * T.nbchildren = len(T.children)
 - Implémentation premier fils-frère droit
 - * B : classe TreeAsBin
 - * B.key
 - * B.child: le premier fils
 - * B.sibling: le frère droit
- Listes : comme d'habitude : len, range, min, max, abs, append
- Les files (les méthodes de la classe Queue, que l'on suppose importée)
 - Queue(): retourne une nouvelle file
 - q.enqueue(x): enfile l'élément x dans q
 - q.dequeue(): supprime et renvoie le premier élément de q
 - q.isempty(): teste si q est vide

Vos fonctions

Vous pouvez également écrire vos propres fonctions à condition qu'elles soient documentées: donnez leurs des spécifications (on doit savoir ce qu'elles font).

Dans tous les cas, la dernière fonction doit être celle qui répond à la question.