

## AL5 Algorithmique Examen

Lundi 14 janvier 2013 12h–15h

Documentation autorisée : une feuille A4 recto verso manuscrite.

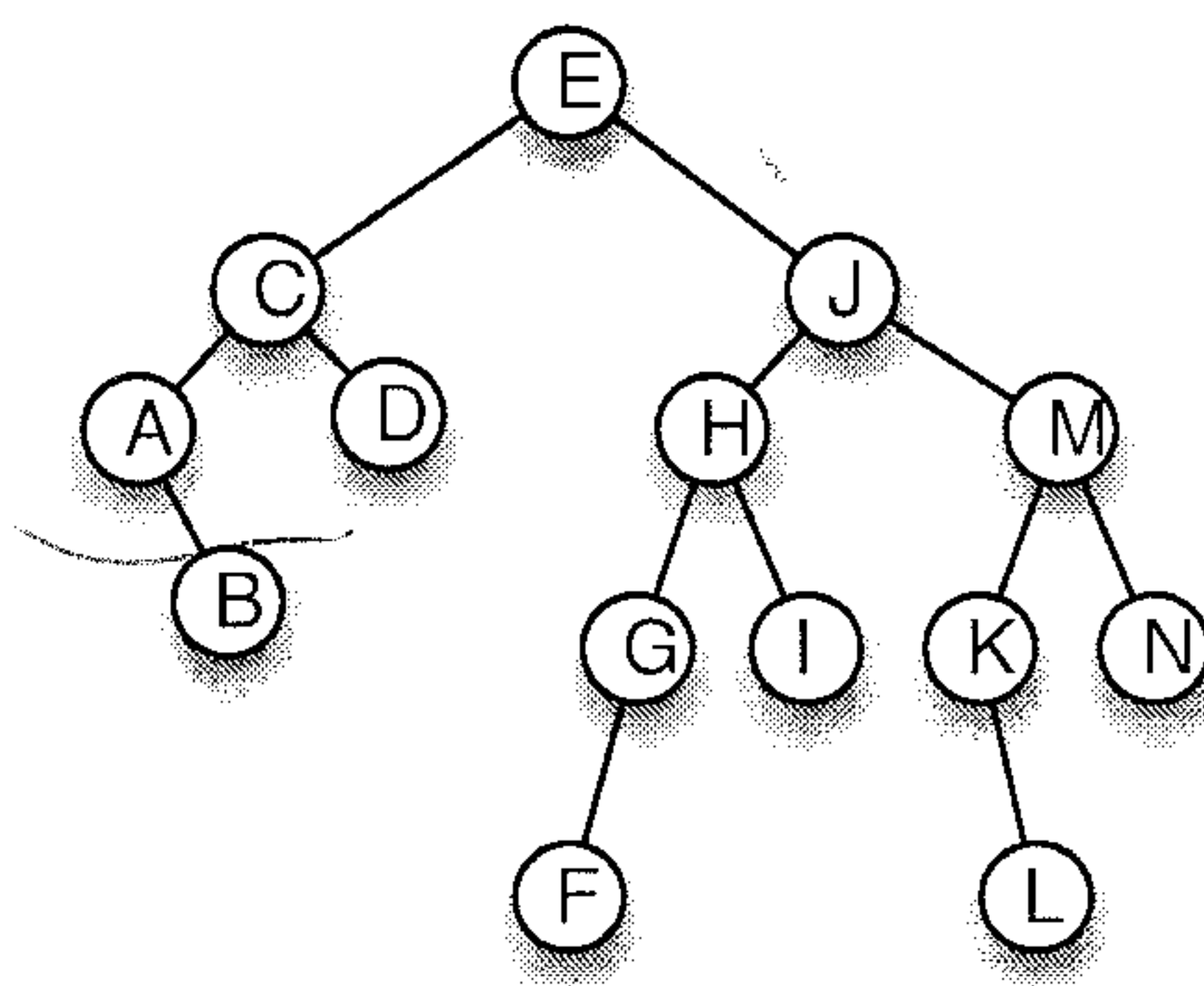
**Indications :** Le barème est donné à titre indicatif seulement. Justifier vos réponses et expliquer vos algorithmes. Les algorithmes doivent être aussi efficaces que possibles. Une partie de la note portera sur la clarté de présentation de vos réponses.

**Exercice 1** (8 points) Le but de cet exercice de concevoir et analyser un algorithme efficace qui prend en entrée un tableau trié d'entiers distincts et qui construit un AVL qui contient les mêmes valeurs que celles dans le tableau. Votre algorithme procédera récursivement : sélectionner d'abord l'élément à placer à la racine, puis construire récursivement les sous-arbres gauche et droite, sans utiliser de rotations. (Un algorithme qui parcourt le tableau et insère les éléments dans l'ordre en utilisant les algorithmes vus en cours sera considéré hors sujet.)

1. Expliquer brièvement l'idée de votre algorithme. Donner l'AVL que votre algorithme construit sur entrée [2,3,5,7,8,10].
2. Donner l'algorithme.
3. Démontrer que si le tableau donné en entrée est trié et contient des valeurs distinctes, alors l'arbre construit par votre algorithme est un AVL.
4. Analyser la complexité de votre algorithme.

**Exercice 2** (2 points) La suppression d'un noeud dans un AVL peut entraîner une cascade de rotations.

1. Identifier un noeud dans l'arbre ci-dessous dont la suppression provoquerait une cascade de suppressions. Identifier la première rotation à effectuer, et expliquer pourquoi elle est nécessaire. *rotations*
2. Donner l'arbre après la première rotation.
3. Expliquer pourquoi une seconde rotation est nécessaire. Donner l'arbre après la seconde rotation. L'arbre est-il équilibré ?



**Exercice 3** (6 points)

1. Donnez un algorithme qui prend en entrée
- un graphe orienté non-valué (les arêtes ont toutes un coût unitaire),
  - un sommet de départ
  - un entier  $d$ ,
- et qui affiche tous les sommets à distance  $d$  du sommet de départ.
2. Analyser la complexité de votre algorithme en fonction de  $n$ , le nombre de sommets dans le graphe.

**Exercice 4** (4 points) Donnez une exécution de l'algorithme Prim et une exécution de l'algorithme Kruskal sur le graphe valué ci-dessous. Pour chacun des deux algorithmes,

1. préciser l'ordre dans lequel les sommets ou les arêtes ont été sélectionnés et
2. donner l'arbre couvrant minimal trouvé par l'algorithme.

