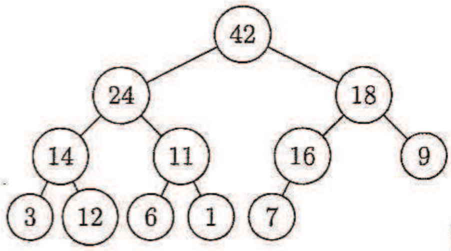
Partiel

Le 23 novembre 2011 – durée 1h55

Notes de cours et de td autorisées. La qualité de la rédaction sera prise en compte dans l'évaluation.

Exercice 1 [Arbres binaires de recherche] Donner un algorithme qui étant donnés un arbre binaire de recherche et deux valeurs a et b, affiche les nœuds étiquetés par des valeurs x tel que $a \leq x \leq b$. La complexité dans le pire des cas devrait être $O(N + h)$, où N est le nombre de nœuds étiquetés par des valeurs entre a et b, et h est la hauteur de l'arbre.

Exercice 2 [Tas] Dans cet exercice, on considère des tas représentés par des tableaux. Etant donné un tas, on remplit le tableau de gauche à droite avec les clés des nœuds de profondeur 0, 1, …, h pris de gauche à droite. Par exemple,



42	24	18	14	11	16	9	3	12	6	1	7			

Les indices des tableaux commencent à 1. Si i est la position d'un nœud ayant un fils droit et un fils gauche, alors $2i$ est la position de son fils gauche et $2i + 1$ est la position de son fils droit. Aussi, si i est la position d'un nœud différent de la racine, alors $\lfloor \frac{i}{2} \rfloor$ est la position de son père.

Question 1. Écrire une procédure pour insérer un nœud dans un tas représenté par un tableau, qui suit la procédure d'insertion donnée en TD pour un tas représenté comme un arbre.

Question 2. Dans la procédure précédente, on fait $O(\log n)$ comparaisons dans le pire des cas, où n est la taille du tas. Donner une nouvelle procédure d'insertion (toujours pour des tas représentés par des tableaux) qui fait $O(\log \log n)$ comparaisons dans le pire des cas.

Attention, on compte seulement le nombre de *comparaisons* : si besoin, on peut déplacer un grand nombre de valeurs lorsque les déplacements ne nécessitent pas de comparaison.

(Indication : on pourra utiliser une recherche dichotomique pour trouver la place du nœud à insérer).

Exercice 3 [Graphes : circuits eulériens]

Vous avez vu en TD une condition nécessaire et suffisante pour qu'il existe un cycle eulérien dans un graphe non orienté. Dans cet exercice nous transposons cette notion aux graphes orientés. On rappelle quelques définitions pour un graphe orienté $G = (V, E)$:

chemin : un chemin C de G est une succession d'arcs adjacents. Autrement dit, $C = ((s_1, d_1), (s_2, d_2), \ldots, (s_n, d_n))$ est un chemin si $(s_i, d_i) \in E$ pour tout $0 < i \leq n$, et $d_i = s_{i+1}$ pour $0 < i < n$.

circuit : un circuit de G est un chemin tel que $d_n = s_1$.

circuit eulérien : un circuit de G est eulérien s'il visite chaque arc du graphe exactement une fois.

graphe équilibré : G est équilibré si le degré entrant de chaque sommet est égal à son degré sortant.

fortement connexe : G est fortement connexe si pour tous sommets x et y, il existe un chemin qui va de x à y et un chemin de y à x.

graphe connexe : G est connexe si deux sommets quelconques sont reliés dans le graphe non orienté construit à partir de G en ignorant le sens des arcs.

sommet isolé : un sommet de G est isolé s'il n'a aucun arc entrant et aucun arc sortant.

Dans la suite, $G = (V, E)$ est un graphe orienté sans sommet isolé.

Question 1. Montrer que si un graphe sans sommet isolé possède un circuit eulérien, alors ce graphe est équilibré et fortement connexe.

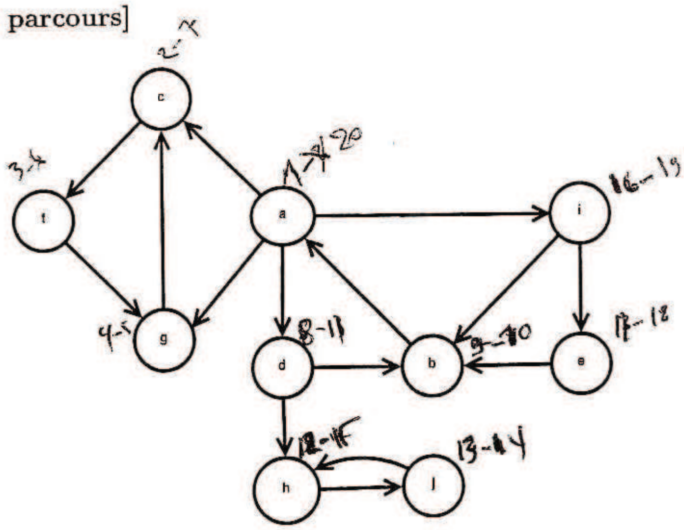
Soit G un graphe équilibré, et soit $C = ((s_1, d_1), (s_2, d_2), \ldots, (s_n, d_n))$ un chemin de G de longueur la plus grande possible, qui n'utilise pas deux fois le même arc.

Question 2. Montrer que $d_n = s_1$, c'est-à-dire que ce chemin est forcément un circuit.

Question 3. Montrer que si C n'est pas eulérien, alors G n'est pas connexe.

Question 4. En déduire une condition nécessaire et suffisante pour qu'un graphe sans sommet isolé possède un circuit eulérien.

Exercice 4 [Graphes : parcours]



Lors de l'exécution des algorithmes on suivra préférentiellement l'ordre alphabétique lorsqu'il y a le choix entre plusieurs sommets possibles ; en particulier on commencera les parcours en a.

Question 1. Dessinez l'arbre donné par le parcours en largeur du graphe précédent.

Question 2. Dessinez l'arbre donné par le parcours en profondeur du graphe précédent.

Redessinez le graphe en indiquant quels sont les arcs du parcours, les arcs retours, les arcs avants et les arcs transverses.

Question 3. Refaire les deux questions précédentes sur la version non orientée du graphe.

Question 4. Montrer que pour un graphe non orienté et connexe, lors de tout parcours en profondeur chaque arête de G est soit une arête du parcours, soit une arête retour.