DEUG MIAS 2e année

TD de programmation fonctionnelle n° 5

Les graphes

Pierre BOULET Jean-Marc NICOD Frédéric VIVIEN

1 Chemin entre deux sommets d'un arbre

On se donne un arbre sous la forme d'une fonction successeur: *succ*. Cette fonction prend en argument la clef (le nombre) associé à un sommet et renvoie la liste des clefs de ses fils. Cette fonction renvoie la liste vide pour un sommet qui n'a pas de fils.

Question 1.1. Écrivez la fonction *succ* pour l'arbre de la figure 1.

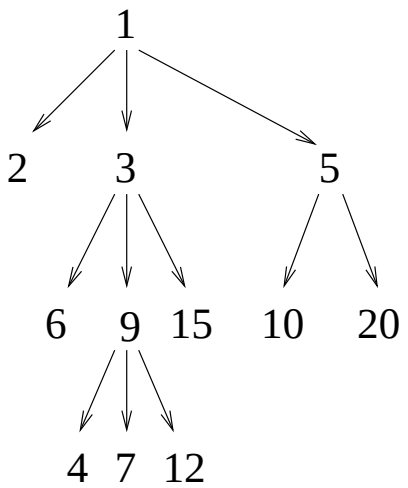

FIG. 1 - *Exemple d'arbre*

Un chemin dans l'arbre est représenté par la suite des sommets traversés.

Question 1.2. Écrivez la fonction *prolonge* qui prend en argument un chemin c représenté par une liste et qui renvoie la liste de tous les chemins obtenus en prolongeant c d'une arête. On supposera que le premier sommet de la liste représentant c est le dernier sommet visité, autrement dit le chemin qui utilise l'arête $1 \rightarrow 3$ est codé (3 1).

Question 1.3. Écrivez une fonction *chemin* qui prend en entrée un sommet *source* et un sommet *but* et qui renvoie le chemin de l'arbre de *source* vers *but* s'il existe, et qui renvoie la liste vide sinon. Écrivez deux versions de cette fonction : une qui réalise un parcours en profondeur et l'autre en largeur.

2 Plus court chemin dans un graphe sans circuits

Nous considérons maintenant des graphes valués.

Question 2.1. Écrivez la fonction *succ* pour le graphe de la figure 2. Ne pas oublier d'associer à chaque successeur la distance qui le sépare de son prédecesseur.

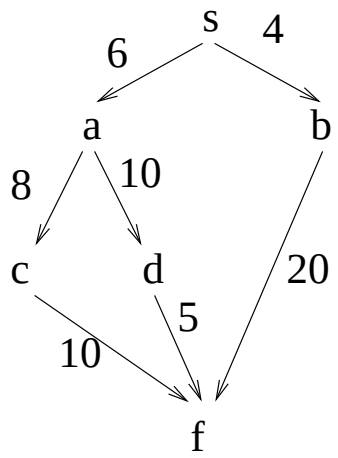

FIG. 2 - *Exemple de graphe valué*

On représente un chemin par une liste. On codera le chemin de s à b de la forme: $((b\ 10)(s\ 0))$, c'est-à-dire en associant à tout sommet du chemin la distance qui le sépare de son prédecesseur.

Question 2.2. Écrivez la fonction *prolonge* qui prend en argument un chemin c représenté par une liste et qui renvoie la liste de tous les chemins obtenus en prolongeant c d'une arête.

Question 2.3. Écrivez la fonction *longueur* qui prend en argument un chemin c et qui renvoie sa longueur.

Question 2.4. Écrivez la fonction *plus-court* qui prend en argument une liste de chemins et qui renvoie le chemin le plus court.

Question 2.5. Écrivez une fonction *chemin* qui prend en entrée un sommet *source* et un sommet *but* et qui renvoie le chemin de l'arbre de *source* vers *but* s'il existe et sa longueur, et qui renvoie la liste vide sinon.

3 Graphe donné en extension: tri topologique

Un graphe est maintenant défini par une liste de listes composées d'un sommet et de ses successeurs. Exemple :

(**define** *Graphe* '((a b f) (b) (c d e) (d a e) (e) (f)))

Question 3.1. Écrivez la fonction *sommets* qui génère la liste des sommets d'un graphe.

Question 3.2. Écrivez la fonction *succ* pour cette représentation.

Question 3.3. Écrivez la fonction *supprimer* qui supprime un sommet d'un graphe.

Question 3.4. Écrivez la fonction *sans-successeurs* qui renvoie la liste des sommets du graphe qui sont sans successeurs.

Question 3.5. Écrivez la fonction *tri-topologique* qui renvoie la liste des sommets du graphe triés dans l'ordre topologique. Un sommet qui n'a pas de successeurs est dernier dans l'ordre topologique.