- Contrôle du 18/12/2007. -

- Durée 1h. Aucun document autorisé. -

- Exercice 1 - Arbre - On se donne un graphe $G = (V, E)$ codé sous forme de liste d'arêtes.
Ecrire un algorithme ARBRECOUVRANT(G) qui retourne FAUX si G n'est pas connexe et l'ensemble des arêtes d'un arbre couvrant sinon.

- Exercice 2 - Un algorithme - On se donne l'algorithme suivant, où l'entrée est un graphe G dont les sommets sont énumérés $v_1, \ldots, v_n$ et la sortie une fonction c définie de l'ensemble des sommets de G dans l'ensemble des entiers strictement positifs.

1. $c(v_1) := 1$
2. Pour i de 2 à n, $c(v_i)$ est le plus petit entier $k > 0$ tel que tous les voisins v_j de v_i dans $\{v_1, \ldots, v_{i-1}\}$ vérifient $c(v_j) \neq k$.
3. Retourner c.

a. Dérouler cet algorithme sur le cube.

b. Quelle propriété vérifie un graphe pour lequel la fonction retournée c est constante égale à 1 ? Même question lorsque les valeurs de c sont inférieures ou égales à 2.

c. Cet algorithme dépend-il de l'ordre des sommets ?

- Exercice 3 - Algorithme de Kruskal -

a. Utiliser l'algorithme de Kruskal pour trouver un arbre couvrant de poids minimum du graphe donné Figure 1. Vous détaillerez les étapes de l'algorithme pour founir l'arbre demandé ainsi que son poids.

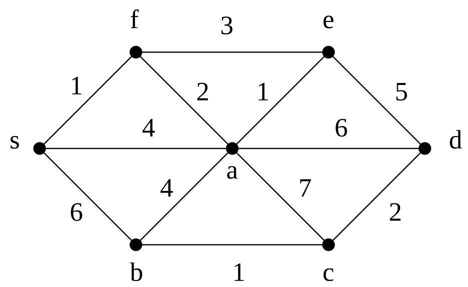

Fig. 1 –

b. Le poids de l'arête bc a été mal calculé, il est en fait de 10. Donner un nouvel arbre couvrant de poids minimum du graphe ainsi modifié, sans utiliser de nouveau l'algorithme de Kruskal.

c. Plus généralement, écrire un algorithme qui prend en entrée un graphe G valué sur ses arêtes, un arbre couvrant de poids minimum T de G, une arête e de T et une valeur k, qui fournit en sortie un arbre couvrant de poids minimum du graphe G dans lequel le poids de e est passé à la valeur k et qui s'exécute en temps $O(m)$, où m est le nombre d'arêtes de G.

d. Prouver la validité de votre algorithme.