TD ALBA - SÉANCE 1
GRAPHES

Exercice 1. Donner le nombre de graphes orientés à n sommets.

Exercice 2. On considère l'algorithme suivant qui prend en entrée un graphe orienté G et qui renvoie une liste topologique de G s'il en admet une et renvoie une erreur si G contient un cycle.

Algorithm 1 Algorithme de liste topologique

$G' \leftarrow G$
$L \leftarrow [\ \ ]$
while $G' \neq \emptyset$ **do**
  Trouver x sommet de G' qui n'est la source d'aucune arête ; sinon renvoyer une erreur.
  $L \leftarrow L \cup \{x\}$
  $G' \leftarrow G' \setminus x$
end while
Renvoyer L.

Montrer que cet algorithme termine et qu'il est correct. On suppose que G est représenté par sa matrice d'adjacence, donner la complexité de cet algorithme.

Exercice 3. On considère le graphe orienté donné par la matrice d'adjacence suivante

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Dessiner le graphe. Montrer qu'il est sans circuit en donnant une liste topologique de ses sommets.

Exercice 4. Soit G un graphe représenté par sa matrice d'adjacence M. Montrer que pour tout $k \geq 0$, le coefficient (i, j) de M^k est égal au nombre de chemins de G allant de i vers j de longueur k.

Exercice 5. On considère le graphe orienté donné par la matrice d'adjacence suivante

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Donner la matrice de la relation d'accessibilité.

Exercice 6. Soit $G = (V, E)$ un graphe orienté à n sommets et m arêtes. Soit x un sommet de G, on appelle *degré sortant* de x le nombre d'arêtes qui partent de x et *degré entrant* de x le nombre d'arêtes qui arrivent en x. On note $\deg_s(x)$ le degré sortant de x et $\deg_e(x)$ le degré entrant de x. Montrer que

(1) $\sum_{x \in V} \deg_e(x) = \sum_{x \in V} \deg_s(x)$.
(2) Montrer que $\sum_{x \in V} \deg_e(x) = m$.

Que peut-on dire si G est non orienté sans boucles ?

Exercice 7. Soit G un graphe orienté connexe, un *circuit eulérien* de G est un circuit qui passe une et une seule fois par chaque arête de G. On dit que G est un graphe eulérien s'il admet un circuit eulérien.

(1) Montrer que si G est eulérien, alors pour tout sommet x de G, $\deg_e(x) = \deg_s(x)$.
(2) Montrer la réciproque.

Quelle est la CNS si G est non-orienté sans boucle ?

Exercice 8. Montrer qu'un graphe non-orienté dont tous les sommets ont des degrés supérieurs ou égaux à 2 a un cycle.

Exercice 9. Soit $G = (V, E)$ un graphe orienté. Pour un chemin $c = (v_1, \ldots, v_r)$ donné on définit l'intérieur $I(c)$ de c comme l'ensemble des sommets du chemin $(v_2, \ldots, v_{r-1})$. On considère la suite de graphes $(G_k)_{0 \leq k \leq n}$ par $G_k = (V, E_k)$ avec : G_0 graphe initial complété d'une boucle en chaque sommet et

$$(i, j) \in E_k \Leftrightarrow \text{ il existe un chemin } c \text{ de } i \text{ vers } j \text{ tel que } I(c) \text{ soit inclus dans } \{1, \ldots, k\}.$$

(1) Montrer par récurrence sur k que la suite de graphe construit dans cet algorithme est la même que celle de l'algorithme de Roy-Warshall.
(2) Montrer que pour tout $i, j, k \in V$ on a $(i, k) \in E_k$ si et seulement si $(i, k) \in E_{k-1}$. De même $(k, j) \in E_k$ si et seulement si $(k, j) \in E_{k-1}$.

Exercice 10. Soit G un graphe non orienté à n sommets ($n \geq 1$).

(1) Montrer que si G est connexe, alors il possède au moins $n - 1$ arêtes.
(2) On dit que G est un *arbre* s'il est connexe et sans cycle. Montrer les équivalences suivantes.
   (a) G est un arbre.
   (b) G ne contient pas de cycle et possède $n - 1$ arêtes.
   (c) G est connexe et possède $n - 1$ arêtes.
   (d) Pour tout sommet u, v de G il existe un unique chemin allant de u à v.
   (e) G est connexe et pour tout arête e de G le graphe $G \setminus e$ n'est plus connexe.
   (f) G est sans cycle mais l'addition d'une arête crée exactement un cycle.