Université de Bourgogne
Examen de Graphes (I6S3)

L3 Informatique
Session de Mai 2024

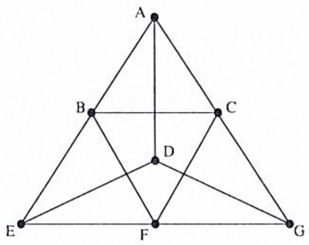
Durée 2h, tous documents autorisés
Sujet recto/verso, le barème est donné à titre indicatif

Exercice 1 (3 pts)

Dessiner tous les graphes connexes a 5 sommets et de degré maximum au plus 3 en les classant suivant la taille du plus grand cycle.

Exercice 2 (5 pts)

Pour le graphe ci-dessous :



1. Proposer un dessin planaire de celui-ci et indiquer le nombre de faces.
2. Déterminer son degré maximum Δ, sa taille de clique maximum ω et sa taille d'ensemble indépendant maximum α.
3. En déduire des encadrements pour son nombre chromatique χ, indice chromatique χ' et nombre chromatique total χ''.
4. Déterminer maintenant exactement χ, χ' et χ'' en donnant pour chaque cas une coloration optimale.
5. Donner un ordre des sommets pour lequel l'algorithme glouton de coloration des sommets est optimal. Exist-t-il un ordre des sommets pour lequel l'algorithme de coloration glouton n'est pas optimal ? Justifier.

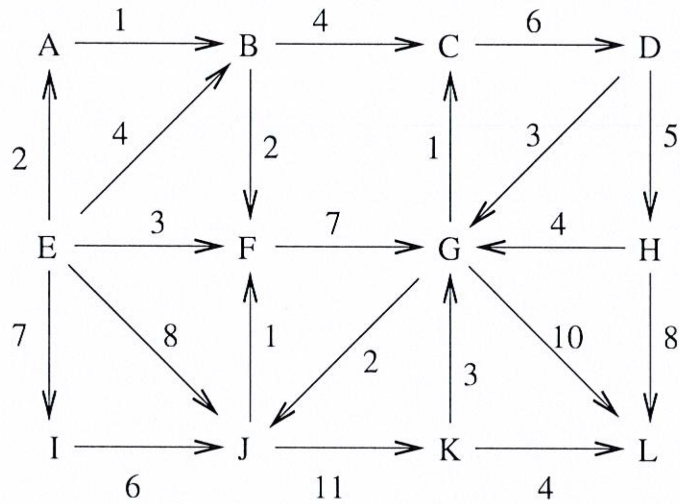
Exercice 3 (4 pts)

Pour un graphe non orienté G, son graphe représentatif des arêtes $L(G)$ est le graphe ayant pour sommets les arêtes de G et deux sommets de $L(G)$ sont adjacents si les arêtes correspondantes de G sont incidentes à un même sommet, c-à-d. tel que $V(L(G)) = E(G)$ et $E(L(G)) = \{ee' |\ e, e' \in E(G) \text{ et } e \text{ et } e' \text{ ont un sommet en commun}\}$.

1. Dessiner le graphe représentatif des arêtes $L(K_4)$ du graphe complet a 4 sommets K_4.
2. Que peut-on dire de la taille de clique maximum de $L(K_4)$? Plus généralement, exprimer $\omega(L(K_n))$ en fonction de n.

Exercice 4 (8 pts)

Soit le graphe orienté pondéré ci-dessous.



1. En utilisant l'algorithme de Ford-Fulkerson, trouver le flot maximum entre E et L. La liste des chaînes augmentantes sera présentée en ordre décroissant de leurs valeurs. Justifier la réponse en exhibant une coupe minimum.
2. En ignorant les valeurs sur les arcs, donner l'ordre de parcours des sommets et l'arbre produit par l'algorithme BFS puis DFS (on supposera que les voisins d'un sommet sont pris par ordre alphabétique) exécuté depuis le sommet E (en suivant le sens des arcs).
3. En considérant les valeurs sur les arcs comme des coûts, déterminer un arbre des plus courts chemins depuis le sommet E avec l'algorithme de Dijkstra. L'évolution des distances vers chaque sommet ainsi que le sommet choisi à chaque étape seront indiqués.
4. En oubliant l'orientation des arcs, déterminer, à l'aide de l'algorithme de Kruskal, un arbre couvrant de poids minimum et indiquer l'ordre d'ajout des arêtes à l'arbre.