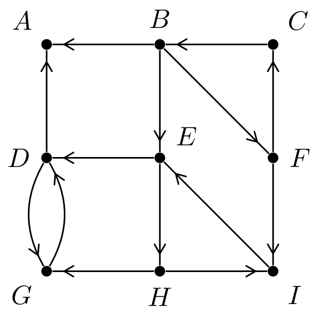
Algorithmique (AL5) **Examen** 6 janvier 2023

Justifier proprement vos réponses ; vous ne recevrez pas tous les points pour une réponse correcte sans justification. On peut énoncer des résultats du cours sans les démontrer. Les documents ne sont pas autorisés à l'exception d'une feuille A4 recto-verso. Les appareils électroniques sont interdits. Le document fait 2 pages et contient 6 exercices. Le barème (sur 20 points) est inscrit à titre indicatif et est susceptible de changements.

Exercice 1. Parcours en profondeur (4 points)

Soit $G = (V, E)$ le graphe orienté ci-dessous, et soit $G^R = (V, E^R)$ le graphe inverse : $(u, v) \in E$ si et seulement si $(v, u) \in E^R$.



Question 1. Exécuter DFS (parcours en profondeur) sur G^R, et indiquer pour chaque sommet u la valeur de $\mathrm{post}(u)$. Chaque fois que vous avez le choix entre plusieurs sommets, choisir celui qui est le premier par ordre alphabétique.

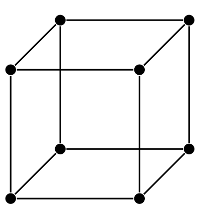
Question 2. Exécuter DFS sur G en traitant les sommets par ordre de $\mathrm{post}(u)$ (trouvé dans la question 1) décroissant, et en déduire les composantes fortement connexes de G.

Question 3. Quelles sont les composantes fortement connexes de G de type source (aucun arc n'entre à cette composante) et de type puits (aucun arc ne sort de cette composante) ?

Question 4. Est-il possible de rendre G fortement connexe en rajoutant un seul arc à G ? Si oui, quel arc ?

Exercice 2. Arbre couvrant de poids minimum (3 points)

Soit G le graphe ci-dessous, où les arêtes horizontales sont de poids 1, les arêtes verticales de poids 2, et les arêtes diagonales de poids 3.

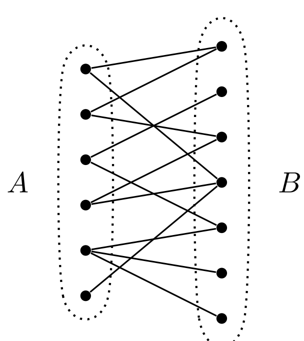


Question 1. Dessiner un arbre couvrant de G de poids minimum et indiquer son poids.

Question 2. Indiquer le nom de l'algorithme que vous avez utilisé. Connaissez-vous un autre algorithme pour résoudre ce problème ? Si oui, indiquer son nom.

Question 3. Combien y a-t-il d'arbres couvrants de poids minimum de G ?

Exercice 3. Flots et couplages (3 points)



Soit G le graphe biparti ci-contre. Le but de cet exercice est de trouver un couplage maximum dans G en utilisant les flots.

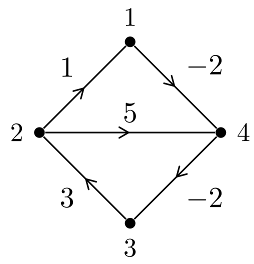
En orientant les arêtes de A vers B et en rajoutant deux sommets s et t ainsi que les arcs de s vers A et les arcs de B vers t, construire un réseau G^*. Tous les arcs dans G^* sont de capacité 1.

Question 1. En utilisant l'algorithme de Ford–Fulkerson, trouver un s–t flot maximum dans G^*. (Donner le flot final ; il n'est pas nécessaire de montrer toutes les étapes.)

Question 2. Vérifier l'optimalité du flot de la question précédente en trouvant une s–t coupe dans G^* de capacité égale à la valeur du flot.

Question 3. Déduire un couplage maximum de G.

Exercice 4. **Plus court chemin entre toutes les paires de sommets (3 points)**



Question 1. En utilisant l'algorithme de Floyd–Warshall, déterminer la distance entre toutes les paires de sommets dans le graphe ci-contre. Pour chaque itération de la boucle principale, donner la matrice des distances.

Question 2. Connaissez-vous un autre algorithme pour résoudre ce problème ? Si oui, indiquer son nom.

Question 3. Quelle propriété de la matrice des distances finale est équivalente à la (non) existence de cycles négatifs dans G ?

Exercice 5. **Vrai ou faux (3 points)**

Répondre vrai ou faux à chacune des questions ci-dessous. Dans cet exercice (et uniquement dans cet exercice), aucune justification n'est demandée.

1. Si G est un graphe avec n sommets et m arêtes, alors $m = O(n)$.
2. Un graphe orienté est un graphe orienté acyclique (DAG) ssi DFS (parcours en profondeur) ne trouve aucun arc transverse.
3. Un graphe orienté à n sommets qui a moins de n composantes fortement connexes n'est pas un DAG.
4. L'algorithme de Bellman–Ford fonctionne uniquement sur les graphes avec poids non négatifs.
5. Si le nombre d'arêtes d'un graphe est impair, alors le graphe contient au moins un sommet dont le degré n'est pas un multiple de 4.
6. Toute arête de poids minimum dans un graphe connexe pondéré appartient à un arbre couvrant de poids minimum.

Exercice 6. **Problème du voyageur de commerce (4 points)**

Vous avez acheté un Pass Interrail, et vous envisagez de visiter les 5 villes suivantes cet été : Amsterdam, Londres, Francfort, Milan et Vienne. Il faut partir de Paris, visiter les 5 villes dans un ordre quelconque, et retourner à Paris à la fin. Les temps de trajet en train entre les différentes paires de villes sont indiqués dans le tableau ci-dessous.

	Amsterdam	Francfort	Londres	Milan	Paris	Vienne
Amsterdam	×	3h45	4h	7h50	3h10	13h35
Francfort	3h45	×	6h25	7h10	3h40	6h20
Londres	4h	6h25	×	10h	2h25	13h25
Milan	7h50	7h10	10h	×	6h30	9h40
Paris	3h10	3h40	2h25	6h30	×	10h30
Vienne	13h35	6h20	13h25	9h40	10h30	×

Pour trouver le meilleur itinéraire, vous décidez d'utiliser l'algorithme de Christofides que vous avez vu dans le cours d'Algorithmique. Soit G le graphe complet à six sommets, dont les sommets correspondent aux villes, et le poids de chaque arête correspond au temps de trajet entre les villes. (*Il n'est pas nécessaire de dessiner le graphe G.*)

Question 1. Trouver un arbre couvrant de poids minimum de G. Quel algorithme avez-vous utilisé ?

Question 2. Déterminer l'ensemble U des sommets de degré impair dans T.

Question 3. Trouver (à la main) un couplage parfait M de poids minimum dans le sous-graphe $G[U]$ induit par les sommets dans U.

Question 4. Déduire un itinéraire pour votre voyage.