Université Clermont Auvergne
BUT Info 1A
R207 - Graphes
Année 2021-2022

Florent Foucaud
Malika More
Thibault Ralet
Carine Simon
Thierry Trévisan

TD1 - Découverte des graphes

Exercice 1 (Trouver tous les petits graphes)**.**
Un graphe (orienté ou non-orienté) est *simple* s'il n'a pas de boucles (une arête ou un arc qui relie un sommet à lui-même) ni d'arêtes multiples (ou d'arcs multiples pour les graphes orientés). Un graphe orienté simple peut quand même avoir un arc de x vers y et un autre arc de y vers x. Un graphe est *connexe* si tout sommet est accessible depuis tout autre (sans considérer le sens des arcs dans le cas des graphes orientés). Un *arbre* est un graphe non orienté connexe et sans cycle.

1. Dessiner tous les graphes connexes non orientés simples à au plus 4 sommets.
2. Dessiner tous les graphes connexes orientés simples à au plus 3 sommets (arcs symétriques et boucles autorisés).
3. Dessiner tous les arbres non orientés à au plus 5 sommets.
4. Comparer le nombre de graphes obtenus à chaque question. Comment expliquer la différence entre ces nombres ?
5. Qu'aurait-on fait si on avait considéré les boucles ?

Exercice 2 (Graphe orienté pour don du sang)**.**

Il existe quatre groupes sanguins :

- AB pour les personnes ayant des antigènes A et B,
- A pour les personnes ayant des antigènes A mais pas d'antigènes B,
- B pour les personnes ayant des antigènes B mais pas d'antigènes A,
- O pour les personnes n'ayant ni antigènes A ni antigènes B.

Il existe également deux rhésus sanguins :

- positif (+),
- négatif (-).

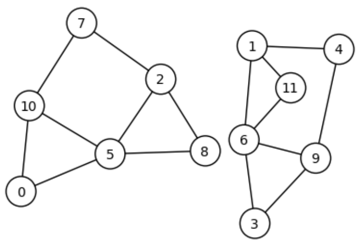
On admet que les seuls interdits biologiques pour recevoir du sang sont les suivants :

- recevoir du sang possédant un antigène dont on est dépourvu,
- recevoir du sang ayant un rhésus positif si on est rhésus négatif.

1. Dessiner le graphe orienté dont $\{AB+, AB-, A+, A-, B+, B-, O+, O-\}$ est l'ensemble des sommets et dont les arcs désignent les possibilités de donner du sang sans enfreindre les interdits biologiques.

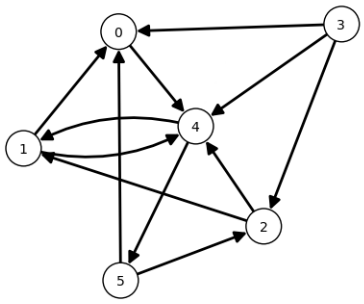
2. Donner le degré sortant $d^+(s)$ (nombre d'arcs qui partent de s) et le degré entrant $d^-(s)$ (nombre d'arcs qui entrent dans s) de chaque sommet s.
3. On appelle *source* un sommet sans voisin entrant (autre que lui-même), *puits* un sommet sans voisin sortant (autre que lui-même) et *sommet universel* un sommet qui a un arc sortant vers tous les sommets. Y a-t-il de tels sommets dans ce graphe ? Si oui, qu'en déduit-on pour les groupes sanguins correspondants ?

Exercice 3 (Matrice d'adjacence)**.**
On considère le graphe (non orienté) suivant.



1. Donnez sa matrice d'adjacence.
2. Donnez les matrices d'adjacence des composantes connexes de ce graphe.
3. À quoi voit-on sur la matrice d'adjacence qu'un graphe n'est pas connexe ? Expliquez comment on peut ordonner les sommets pour que ce soit encore plus flagrant pour le graphe ci-dessus.

Exercice 4 (Matrice d'adjacence de graphe orienté)**.**
On considère le graphe orienté suivant.



1. Donnez sa matrice d'adjacence.
2. À quoi voit-on sur la matrice d'adjacence qu'un graphe est non orienté ?

Exercice 5 (Nombre d'arêtes et degrés dans un graphe)**.**
Le *graphe biparti complet* $K_{a,b}$ a deux ensembles de sommets A et B de taille a et b, avec une arête entre chaque sommet de A et chaque sommet de B (mais aucune autre arête). Le *graphe complet* K_n est le graphe à n sommets avec toutes les arêtes possibles. Le *degré* d'un sommet est le nombre de ses voisins.

1. Quel est le nombre d'arêtes d'un graphe biparti complet $K_{a,b}$ avec a et b sommets dans chaque partie ? Combien cela fait-il lorsque $a = b = n/2$?
2. Quel est le nombre d'arêtes du graphe complet à n sommets, K_n ?

3. Quelle relation relie le nombre d'arêtes d'un graphe non orienté et la somme des degrés de tous les sommets ?
4. Le *principe des tiroirs* énonce que si on a n paires de chaussettes rangées dans m tiroirs, si $m<n$, alors deux paires se retrouvent nécessairement dans le même tiroir.

   On peut en déduire des faits amusants, par exemple, dans le département du Puy de Dôme, il existe au moins deux personnes avec le même nombre de cheveux sur la tête (car le nombre maximum de cheveux sur la tête d'un être humain est environ 500000, et la population du Puy de Dôme environ 650000).

   Déduire du principe des tiroirs que pour un groupe G de personnes dans lequel chacun a au moins un ami dans G (autre que lui-même), il existe au moins deux personnes de G qui ont le même nombre d'amis parmi les personnes de G. *(Indice : modéliser la situation par un graphe.)*
5. (**) Le *principe des tiroirs généralisé* énonce que si on a n paires de chaussettes rangées dans m tiroirs, alors au moins un tiroir contient au moins $\lceil n/m \rceil$ paires de chaussettes. Utiliser ce principe pour montrer que parmi tout groupe de 6 personnes, il existe 3 personnes qui soit se connaissent toutes les trois, soit ne se connaissent pas du tout. *(Indice : considérer n'importe quelle personne et définir deux tiroirs.)*
6. Montrer avec un exemple, que l'énoncé de la question précédente n'est pas forcément vrai pour tout groupe de 5 personnes.

Exercice 6. On considère le code SQL ci-dessous.

```
CREATE TABLE tfamille
(codefamille CHAR(6) PRIMARY KEY,
nom VARCHAR2(30),
CONSTRAINT u_nom_famille UNIQUE (nom));

CREATE TABLE tclient
(referenceclient CHAR(6) PRIMARY KEY,
nom VARCHAR2(20),
rue VARCHAR2(30),
codepostal CHAR(5),
ville VARCHAR2(20),
typeclient CHAR(1),
typetarif CHAR(1),
CONSTRAINT ck_code_typeclient
   CHECK (typeclient='0' OR typeclient='1' OR typeclient='2' OR typeclient='3'),
CONSTRAINT ck_typeclient_typetarif
   CHECK ((typetarif='G' AND
          (typeclient='1' OR typeclient='2' OR typeclient='3')) OR typetarif='N'));

CREATE TABLE tproduit
(referenceproduit CHAR(6) PRIMARY KEY,
designation VARCHAR2(30),
stock NUMBER(6),
codeetat CHAR(1),
prixnormal NUMBER(8,2),
prixgrossiste NUMBER(8,2),
codefamille CHAR(6) NOT NULL,
CONSTRAINT fk_tproduit_tfamille
   FOREIGN KEY (codefamille) REFERENCES tfamille(codefamille),
CONSTRAINT ck_prixnormal_prixgrossiste
```

```
   CHECK (prixgrossiste <= prixnormal),
CONSTRAINT ck_stock_non_negatif
   CHECK (stock >0 OR stock =0),
CONSTRAINT ck_codeetat_valeurstock
   CHECK ((codeetat='E' AND stock=0) OR codeetat='D'));

CREATE TABLE tvente
(numerovente CHAR(6) PRIMARY KEY,
datevente DATE,
referenceclient CHAR(6) NOT NULL,
CONSTRAINT fk_tvente_tclient
   FOREIGN KEY (referenceclient) REFERENCES tclient(referenceclient));

CREATE TABLE tventeproduit
(numerovente CHAR(6),
referenceproduit CHAR(6),
quantite NUMBER(6),
prixventeunitaire NUMBER(8,2),
CONSTRAINT fk_tventeproduit_tproduit
   FOREIGN KEY (referenceproduit) REFERENCES tproduit(referenceproduit),
CONSTRAINT fk_tventeproduit_tvente
   FOREIGN KEY (numerovente) REFERENCES tvente(numerovente),
CONSTRAINT pk_tventeproduit PRIMARY KEY (referenceproduit,numerovente));
```

1. Expliquez comment on peut construire un graphe orienté $\vec{G}$ à partir de ce code, de sorte que $\vec{G}$ facilite la génération de nouveau code permettant de supprimer des données des tables.
2. Identifiez une propriété du graphe $\vec{G}$ qui modélise le cas où l'on ne peut pas forcément supprimer les données en utilisant seulement une séquence de commandes de la forme `DELETE * FROM ma_table`.

Exercice 7 (Tournois)**.**
On fait jouer des équipes sportives entre elles de façon à ce que chaque équipe rencontre chaque autre équipe. On modélise le résultat par un graphe orienté, avec un sommet pour chaque équipe et un arc d'une équipe A à une équipe B si A a gagné contre B. Ce type de graphe orienté est appelé un *tournoi*. Un tournoi est dit *transitif* si, pour tout ensemble de trois équipes A,B,C telle que A a battu B et B a battu C, alors A a aussi battu C.

1. Donner un exemple de tournoi transitif et un exemple de tournoi non transitif, tous les deux à 4 sommets.
2. Montrer que dans un tournoi transitif, il y a une équipe qui a perdu tous les matches et une équipe qui les a tous gagnés.
3. Montrer qu'un tournoi est transitif si et seulement si il n'a pas de cycle orienté.
4. On appelle *roi* un sommet d'un tournoi qui a un chemin orienté de longueur au plus 2 vers tous les autres sommets. Trouver un tournoi à 5 sommets où tout sommet est un roi.
5. Montrer que tout tournoi possède au moins un roi. *(Un indice : choisir un sommet qui a le plus de voisins sortants.)*