L3-INFO – INF6B
Structures discrètes
2021-2022

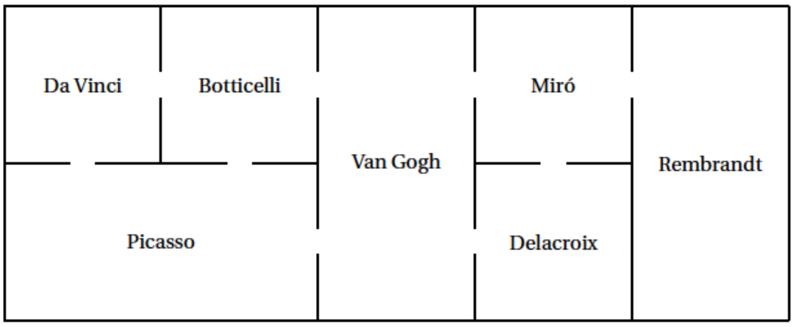
TD 09 – Parcours Eulériens / Parcours Hamiltoniens

Exo. 1

On construit un musée dont les pièces sont disposées comme indiqué sur la figure 2.2 de la présente page (les entrées et sorties du musée ne sont pas crées).

1. Montrer qu'il est impossible d'organiser un parcours dans ce musée qui emprunterait une et une seule fois chaque passage entre deux salles.
2. Quel passage doit-on condamner, ou quel passage doit-on créer pour qu'un tel trajet soit possible ? Dans quelle(s) pièce(s) doit-on alors créer l'entrée et la sortie du musée ?

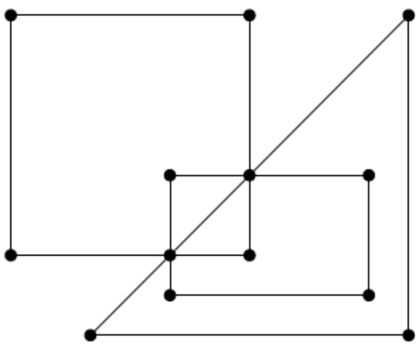
FIGURE 2.2: Figure de l'exercice 2.2



Exo. 2

Pour le graphe de la Figure ci-dessous,

- existe-il un parcours eulérien ? si oui donnez un exemple ;
- sinon expliquez pourquoi.

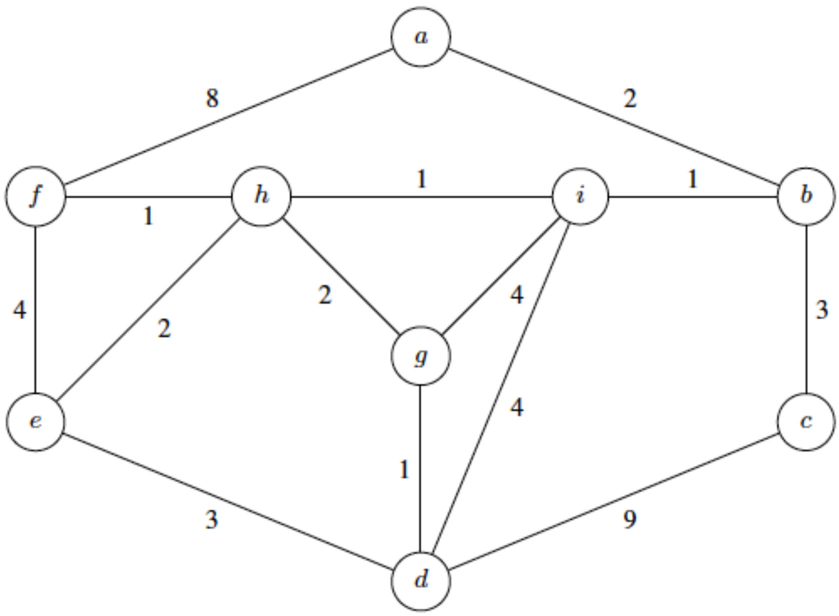


Exo. 3 *(Tour de France des régions)*

La carte de la figure 3 représente les différentes régions administratives Française métropolitaine.

1. Est-il possible de parcourir la France en passant une et une seule fois par toutes les frontières entre toutes les régions ?
2. Si oui, donnez un chemin possible ;
3. Si non, expliquez pourquoi.

Exo. 4 Constater l'absence de parcours Eulérien, puis résoudre le problème du postier chinois ci-dessous :



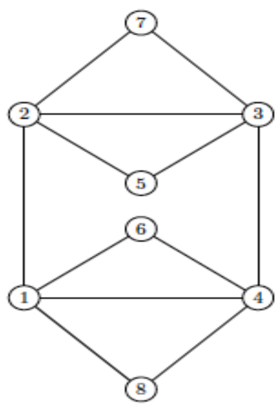
Exo. 5 Algorithme de construction d'un parcours Eulérien (cycle/chaîne).

1. Soit G un graphe dont ***tous les sommets sont de degré pair.*** On souhaite construire un cycle eulérien de G en utilisant l'algorithme ci-dessous :

Algorithm 1 CYCLEEULÉRIEN(G, x)

Require: un graphe G sans sommet de degré impair et x un sommet de G.
Ensure: une liste $(x, x_2, \ldots, x_k, x)$ de sommets de G formant un cycle eulérien.
1: $A \leftarrow$ ensemble des sommets adjacents à x dans G.
2: **if** $A = \emptyset$ **then**
3:    **return** (x)
4: **else**
5:    $C = (x = y_1, \ldots, y_\ell = x) \leftarrow$ un cycle quelconque d'origine x dans G.
6:    supprimer les arêtes de C dans G.
7:    $R \leftarrow ()$
8:    **for** $i = 1$ à ℓ **do**
9:       $R \leftarrow R \cdot$ CYCLEEULÉRIEN(G, y_i)
10:    **end for**
11:    **return** R
12: **end if**

a. On souhaite appliquer cet algorithme au graphe G ci-dessous avec comme sommet initial le sommet x = 1. On suppose que le cycle C obtenu (ligne 5) vaut : $C = 1 - 2 - 3 - 4 - 1$. Dérouler la suite de l'algorithme.



b. Pourquoi cet algorithme est-il correct ? En particulier, pourquoi peut-on faire un appel récursif à la fonction de calcul d'un cycle (ligne #9)

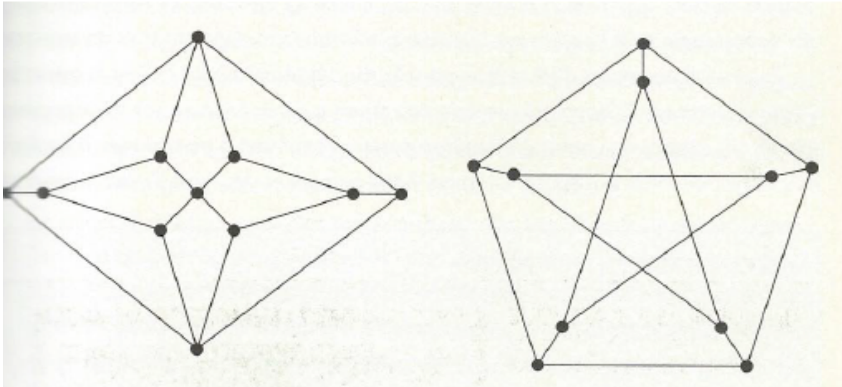
2. Soit G un graphe possédant **(exactement) 2 sommets de degré impair notés *x* et *y*.** *En s'inspirant de l'algorithme précédent,* proposer un algorithme permettant de construire une chaîne eulérienne de G d'origine x et d'arrivée y.

Pour cela, compléter l'algorithme suivant et donner un exemple d'exécution.

Algorithm 2 ChaineEulérienne *(G, x, y)*

Require : un graphe G ayant exactement deux sommets de degré impair x et y.
Ensure : une liste *(x, x_2, …, x_k, y)* de sommets de G formant une chaîne eulérienne.
1. *C = (x = y_1, …, y_l=y)* une chaîne quelconque d'origine x et d'arrivée y dans G.
2. Supprimer les arêtes de C dans G ;
3. *R ← ()* **…/…**

Exo. 6 Les graphes ci-dessous possèdent-ils un parcours hamiltonien ?



Graphe #1 Graphe #2

Exo. 7 Modélisation sous forme de PVC d'un problème de fabrication

Un atelier d'électronique doit produire une série de circuits imprimés tous identiques. Chaque plaque de circuit imprimé doit être percée de *n* trous *P_i (1 ≤ i ≤ n)* définis par leurs coordonnées *(x_i, y_i)* dans un repère orthonormé *(O, x, y)* relatif à la plaque. Les plaques arrivent en file (i.e. les unes après les autres) via une courroie transporteuse qui s'arrête le temps nécessaire sous une machine de perçage. La tête de perçage part du point origine *O*, effectue les *n* trous, puis revient à son point de départ. Elle se déplace à une vitesse de *v* cms par seconde et perce un trou en *p* secondes. Le responsable d'atelier souhaite déterminer l'ordre de perçage des *n* trous afin de minimiser la durée de traitement d'une plaque.

Modéliser ce problème sous forme d'un PVC (on précisera : sommets, arêtes et fonction de poids) en supposant que la tête de perçage peut se déplacer (a) dans n'importe quelle direction (b) uniquement horizontalement et verticalement

Exo. 8 Heuristique du plus proche voisin pour le PVC

```
données un graphe non orienté G = (V, U) ;
début
        soit v_i le sommet de départ ;
        S :={v_i} ;             % on part d'une solution initiale réduite à un seul sommet
        tantque (S ≠ V)         % tant que la solution partielle n'est pas "complète"
                choisir le sommet v_k non visité le plus proche
                S := S U {v_k};         % extension de la solution partielle
        fttque ;
        résultat S U {v_i}
fin
```

- Montrer que la stratégie gloutonne du plus proche voisin calcule bien un circuit hamiltonien de poids minimal sur un exemple (on prendra comme source le sommet a).

- En modifiant le poids d'un arc, montrer que cette stratégie ne donne pas à tout coup une solution optimale. Donner un exemple.

- Complexité en fonction du nombre n de sommets et du nombre m d'arêtes