L3 Informatique • IF05X040 Algorithmique avancée

TP 0

A Premières réflexions sur les représentation de problèmes par graphes

Exercice 1

On souhaite constituer 5 groupes de travail A, B, C, D, E.

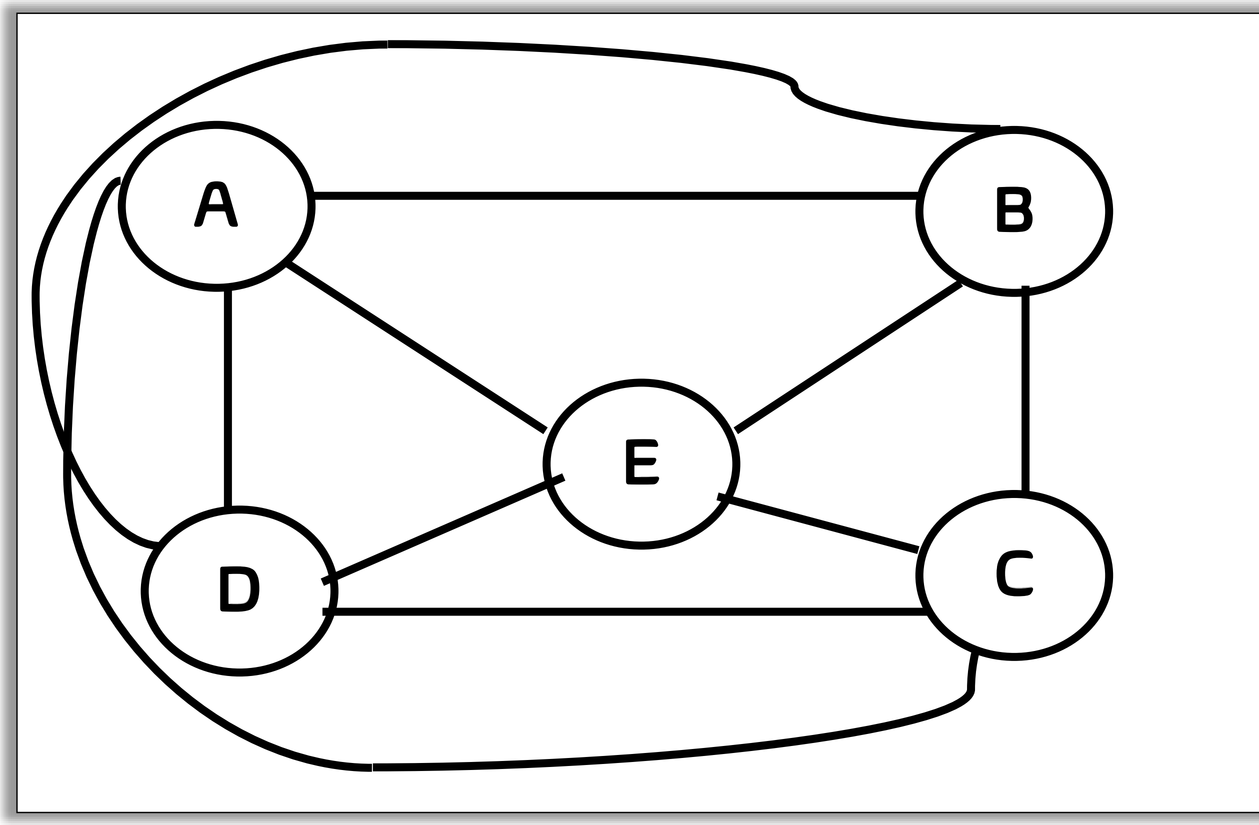
- Deux groupes distincts devront forcément avoir un et un seul membre en commun.
- Une même personne fera partie d'exactement deux groupes de travail.

Combien de personnes doit comporter chaque groupe de travail ?

Combien faut-il de personnes au total ?

- 1er chose : utiliser un sommet pour représenter un groupe.
- 1 arrête va représenter un membre.
- 1 arrête qui relie 2 groupes veut dire qu'une personne appartient à ces 2 groupes.

→ *Les sommet peuvent être placer ou on veut, dans l'ordre qu'on veut, il n'y a pas de règle..*
→ *Ce n'est pas grave si les arrêtes se croisent.*



Quand on a obtenu ce graphe,

on constate qu'il y a 4 arrêtes par groupe.

→ 4 membres différents par groupe.

Quelle genre de graph on a obtenu ?

On a un graphe complet.

Graphe complet

En théorie des graphes, un graphe complet est un graphe simple dont tous les sommets sont adjacents deux à deux, c'est-à-dire que

Tout couple de sommets disjoints est relié par une arête

Nombre de sommets	n
Nombre d'arêtes	$n(n-1)/2$
Distribution des degrés	*(n-1)*-régulier

Pour chaque sommet, son degré est de $n - 1$,
ou n est le nombre de sommets.

$\rightarrow$ Ici $n = 5$, donc il ya 4 personnes par groupe.

Dans cet exercice, le nombre d'arêtes nous donne le nombre de personne en total.

Pour un graphe complet, le nombre total d'arêtes est donné par :

$$\frac{n(n-1)}{2} = \frac{5 \cdot 4}{2} = \frac{20}{2} = 10$$

Exercice 2

Résolution d'un sudoku

❒ Le jeu du Sudoku consiste à compléter une grille de 9 carrés de 3×3 cases chacun,

→ En attribuant à chaque case un chiffre (de 1 à 9)

→ De telle sorte qu'un même chiffre n'apparaisse qu'une et une seule fois sur chaque ligne, sur chaque colonne et dans chaque carré de la grille.

Formaliser le problème sous forme de graphe.

A quelle problématique correspond la résolution d'une grille de sudoku ?
Quelle est l'ordre du graphe et sa taille?

Les cases grises – les case ou on peut plus écrire 1

1								

Formaliser le problème sous forme de graphe.

Pour formaliser la résolution d'un Sudoku sous forme de graphe, on peut utiliser un graphe simple non orienté.

- **On peut prendre pour sommet les cases.**
- **Les arrêtes représentent la relation « ne doit pas contenir le même chiffre » .**
  - → Chaque sommet sera relié aux autres sommets qui font partie de la même ligne, la même colonne, le même carré.
    - Les sommets adjacents sont donc les sommets qui appartiennent à la même ligne, même colonne et même carré.

Quelle est l'ordre du graphe et sa taille ?

1 sommet a 20 sommets qui lui sont adjacents. 81 sommets en total.

Quelle problématique correspond la résolution d'une grille de sudoku ?

En formalisent ainsi le problème,
on voit que la résolution d'une grille de Sudoku est un problème de coloration.
(au lieu d'affecter des couleurs -> affecter des valeurs).

Exercice 3 (1)

Coloration de cartes

On souhaite colorier une carte de telle sorte que deux pays frontaliers n'aient jamais la même couleur.
C'est un problème de coloration de graphe.

Modéliser le problème par un graphe

Que représentent les sommets ?
Quelle est la relation d'adjacence entre deux sommets ?

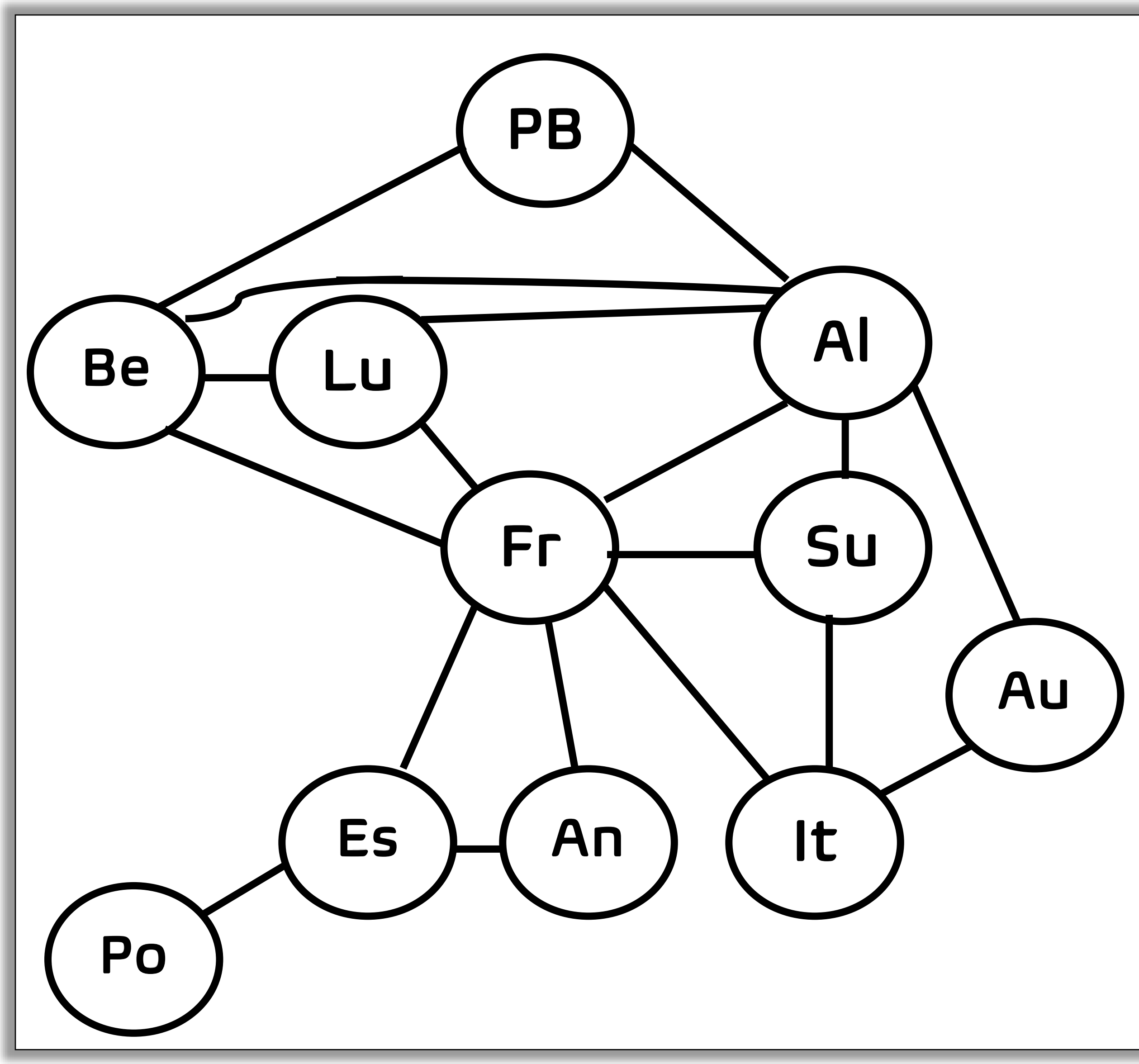
- Les sommets représentent les pays.
- La relation d'adjacence représente le fait que 2 pays soient frontalier.

Exercice 3 (2)

Voici une liste de 11 pays européens
et pour chacun d'eux les pays de cette liste qui leur sont frontaliers

- **France (Fr) :** Espagne, Andorre, Italie, Suisse, Allemagne, Luxembourg, Belgique
- **Espagne (Es) :** France, Andorre, Portugal
- **Portugal (Po) :** Espagne
- **Andorre (An) :** Espagne, France
- **Italie (It) :** France, Suisse, Autriche
- **Autriche (Au) :** Italie, Suisse, Allemagne
- **Suisse (Su) :** France, Italie, Autriche, Allemagne
- **Allemagne (Al) :** France, Suisse, Autriche, Luxembourg, Belgique, Pays-Bas
- **Luxembourg (Lu) :** France, Belgique, Allemagne
- **Belgique (Be) :** France, Luxembourg, Allemagne, Pays-Bas
- **Pays-Bas (PB) :** Belgique, Allemagn

Dessiner le graphe correspondant au problème de coloration



Construire la matrice d'adjacence correspondante

Soit A la matrice d'adjacence.

On met 1 quand deux pays sont frontaliers, 0 sinons.

$$A = \begin{array}{c c} & \begin{array}{ccccccccccc} \text{Fr} & \text{Es} & \text{Po} & \text{An} & \text{It} & \text{Au} & \text{Su} & \text{Al} & \text{Lu} & \text{Be} & \text{PB} \end{array} \\ \begin{array}{c} \text{Fr} \\ \text{Es} \\ \text{Po} \\ \text{An} \\ \text{It} \\ \text{Au} \\ \text{Su} \\ \text{Al} \\ \text{Lu} \\ \text{Be} \\ \text{PB} \end{array} & \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

Appliquer l'algorithme de Welsh et Powell pour trouver une coloration propre.

Sommets	Degré	Rouge	Bleu	Vert	Jaune
Fr	7	Rouge	-	-	-
Al	6	-	Bleu	-	-
Su	4	-	-	Vert	-
Be	4	-	-	Vert	-
Es	3	-	Bleu	-	-
It	3	-	Bleu	-	-
Au	3	Rouge	-	-	-
Lu	3	-	-	-	-
An	2	-	-	Vert	Jaune
PB	2	Rouge	-	-	-
Po	1	Rouge	-	-	-

L'algorithme de Welsh et Powel donne

$\{\boldsymbol{Fr}, \boldsymbol{Aut}, \boldsymbol{PB}, \boldsymbol{Po}\}$ ***peuvent etre de la meme couleur***
$\{\boldsymbol{Al}, \boldsymbol{Es}, \boldsymbol{It}\}$
$\{\boldsymbol{Be}, \boldsymbol{Su}, \boldsymbol{And}\}$
$\{\boldsymbol{Lu}\}$

L'algorithme fournit-il le nombre chromatique ?

Nombre chromatique – nombre de couleur minimum à utiliser pour colorier ce graphe.

On a 4 couleurs pour colorier ce graphe.

Le nombre chromatique est supérieur ou égal au nombre de sommet du plus grand sous-graphe complet.

Le plus grand sous-graphe complet est donnée par $\{Fr, Bel, Lux, Al\}$.

Il contient 4 sommets donc on a bien le nombre chromatique.

B Structure de données

TP12 réponse à la question (d)

```
**** La racine de l'arbre ****
Caractère = ? B

****** Menu ******
G - Insérer fils gauche
D - Insérer fils droit
P - Parcours prefixe
Q - Quitter
Votre choix ? G
** Inserer un nouveau noeud APRES la noeud N (fils gauche) **
N (N > 0) = ? 1
Caractère = ? A

****** Menu ******
G - Insérer fils gauche
D - Insérer fils droit
P - Parcours prefixe
Q - Quitter
Votre choix ? D
** Inserer un nouveau noeud APRES la noeud N (fils droit) **
N (N > 0) = ? 1
Caractère = ? O

****** Menu ******
G - Insérer fils gauche
D - Insérer fils droit
P - Parcours prefixe
Q - Quitter
Votre choix ? P
 B  A  O
****** Menu ******
G - Insérer fils gauche
D - Insérer fils droit
P - Parcours prefixe
Q - Quitter
Votre choix ? G
** Inserer un nouveau noeud APRES la noeud N (fils gauche) **
N (N > 0) = ? 1
Caractère = ? R

****** Menu ******
G - Insérer fils gauche
D - Insérer fils droit
P - Parcours prefixe
Q - Quitter
Votre choix ? P
 B  R  A  O
****** Menu ******
G - Insérer fils gauche
D - Insérer fils droit
P - Parcours prefixe
Q - Quitter
Votre choix ? D
** Inserer un nouveau noeud APRES la noeud N (fils droit) **
N (N > 0) = ? 4
Caractère = ? V

****** Menu ******
G - Insérer fils gauche
D - Insérer fils droit
P - Parcours prefixe
Q - Quitter
Votre choix ? P
 B  R  A  V  O
****** Menu ******
```

Comment étendre la structure de données choisies pour un arbre générique ? Un graphe quelconque ?

- Une arbre générique peut être représenté à l'aide d'une structure de données « nœud » qui comprend un champ qui est un pointeur vers le début d'une liste chaînée qui comprendra tous les fils d'un nœud.

```
typedef struct cellule{

    struct cellule* celPrecedente;
    // Pointeur vers la cellule précédente de la liste.

    struct cellule* celSuivante;
    // Pointeur vers la cellule suivante de la liste.

    char etiquette;
    int numeroDeCreation;

} cellule ;

typedef struct noeud{

    struct noeud* pere;
    // Pointeur vers le pere.

    cellule* fils;

    char etiquette;

    int numeroDeCreation;

} noeud ;
```

- Une graphe quelconque peut être représenté à l'aide d'une :
    - Matrice d'adjacence
    - Liste d’arêtes
    - Liste de voisinages

Source :
cours du Prof. Etienne Birmelé - page 8 - « Codage d’un graphe »

C Activités supplémentaires

Exercice 1

Algorithme de construction de graphe planaire

Soit l’algorithme de construction de graphe planaire suivant
(voir animation sur Planarity.net)

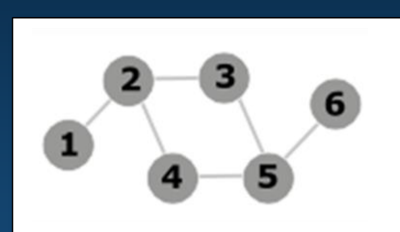
```
fonction G = planaire(G)
tant que G n’est pas planaire faire
        pour tout x sommet de G faire
                V = liste des voisins de x
                M = barycentre des sommets de V
                si on diminue le nombre d’intersections d’arcs
                        alors placer x en M
                fin si
        fin faire
fin faire
```

Quelle est la complexité de cet algorithme ?

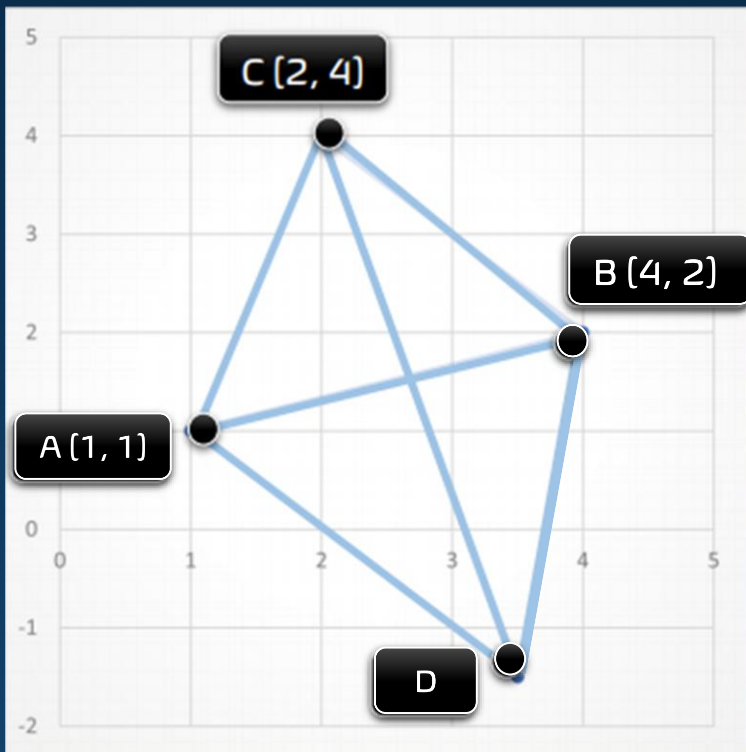
Finit-il toujours ?

Graphe planaire

Dans la théorie des graphes, un graphe planaire est un graphe qui a la particularité de pouvoir se représenter sur un plan sans qu'aucune arête (ou arc pour un graphe orienté) n'en croise une autre



Exemple



x sommet de G

Le sommet D

V = liste des voisins de x

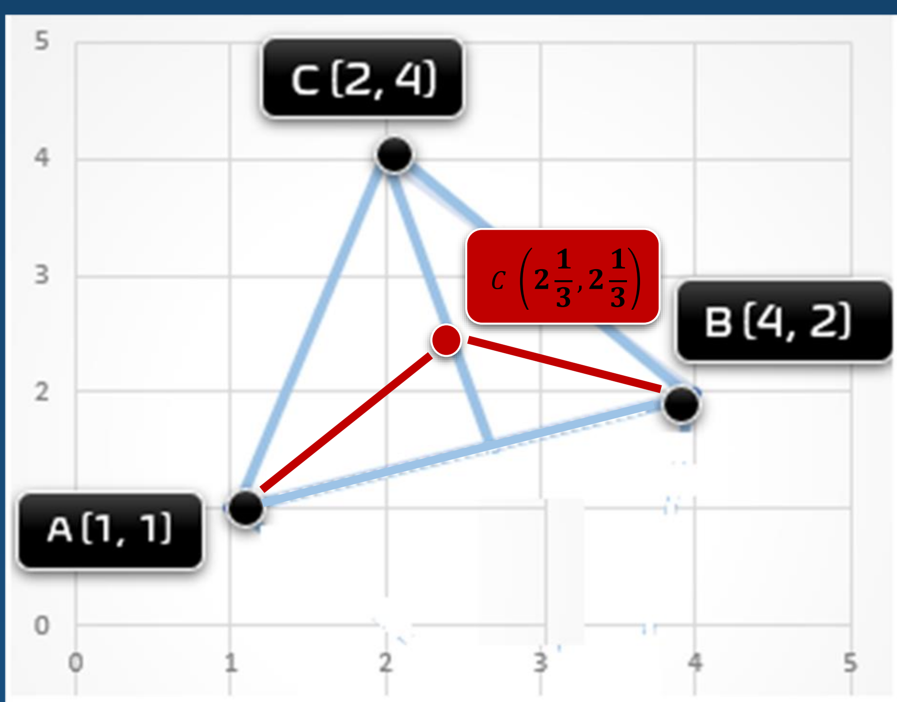
Les voisin de D : (A, B, C)

M = barycentre des sommets de V

$$Br = \left(\frac{1+4+2}{3}, \frac{1+2+4}{3}\right) = \left(\frac{7}{3}, \frac{7}{3}\right)$$

$$Br = \left(\frac{7}{3}, \frac{7}{3}\right) = \left(2\frac{1}{3}, 2\frac{1}{3}\right)$$

Placer x en M diminue le nombre d'intersections d'arcs ?



Finit-il toujours ?

Non.

La condition d'arrêt de la boucle est :

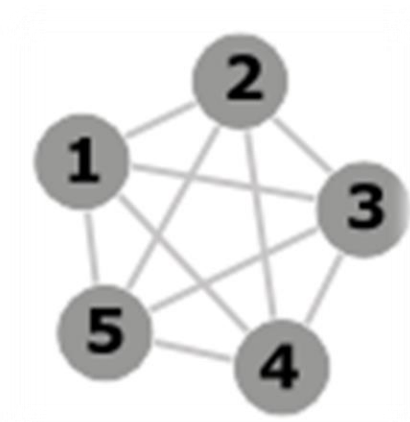
tant que G n'est pas planaire faire

or, pas toujours on peut se ramener a un graphe planaire a partir d'un graphe quelconque :

> Un graphe fini est planaire ***si et seulement s'il*** ne contient pas de sous-graphe partiel qui est une expansion de K5 (le graphe complet à 5 sommets) ou K3,3 (le graphe complet biparti à 3+3 sommets).
>
> **Source : Wikipedia - CC BY-SA 3.0** https://fr.wikipedia.org/wiki/ Graphe_planaire

Par exemple,

Source : Wikipedia - CC BY-SA 3.0 https://fr.wikipedia.org/wiki/ Graphe_planaire



C'est un graphe complet à 5 sommets (K5).

Il n'est pas planaire.

Dans un tel cas, le programme ne se termine jamais.