Introduction à l'informatique – Graphes II

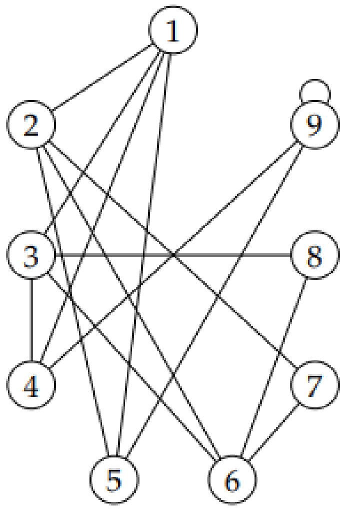
Table des matières

RAPPELS

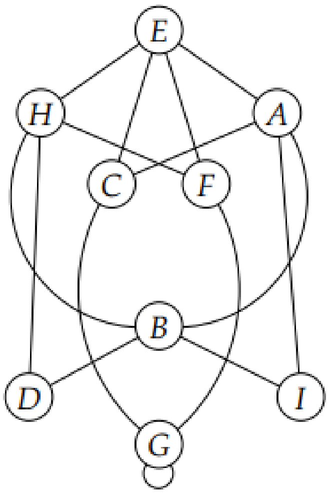
- Soit G = (S, A) un graphe non orienté et a = {u, v} ∈A une de ses arêtes. A est incidente à u et v. u et v sont voisins et adjacents.
- Soit G = (S, A) un graphe orienté et a = (u, v) ∈A une de ses arcs. A est incident à u et à v. u et v sont adjecents. V(resp. u) est voisin sortant (resp. entrant) de u (resp. v) et tel que v ∈ |+ (u) (resp. u ∈ |- (v)) -> diapo page 4
- Soit G = (S, A) un graphe. La matrice adjacente de G, notée B(G), est la matrice booléenne d'ordre |V| tq B(G)_(i,j) = {1 si {i, j} ou (i, j) ∈A | 0 sinon}

DEFINITIONS CLASSIQUES

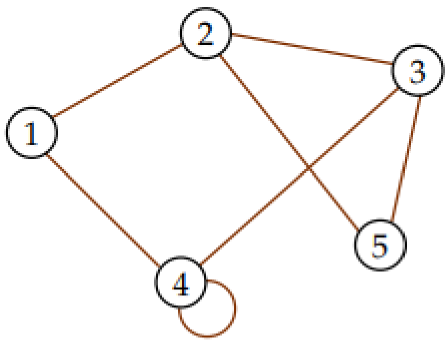
- Deux graphes G = (S, A) et G' = (S', A') sont isomorphes s'il existe une bijection ϕ : S -> S' telle que (u, v) ∈A ⇔ (ϕ (u), ϕ (v)) ∈A'



$$\begin{aligned}\phi(1) &= E\\ \phi(2) &= H\\ \phi(3) &= A\\ \phi(4) &= C\\ \phi(5) &= F\\ \phi(6) &= B\\ \phi(7) &= D\\ \phi(8) &= I\\ \phi(9) &= G\end{aligned}$$

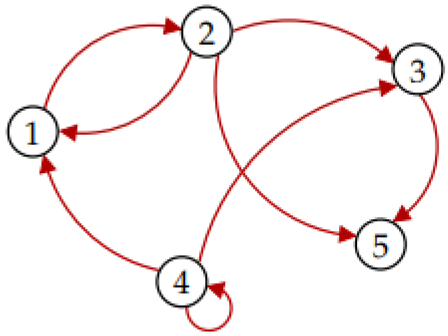


- Soit G = (S, A) un graphe non orienté et s ∈S. Le degré (ou valence) de s, noté d(s) est le nombre d'aretes incidentes à s, les boucles étant comptées deux fois, et telle que Somme_(s ∈S) deg(s) = 2*|a|



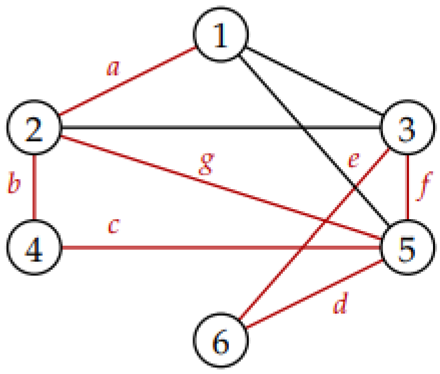
$$d(1) = 2$$
$$d(2) = 3$$
$$d(3) = 3$$
$$d(4) = 4$$
$$d(5) = 2$$

- Soit G = (S, A) un graphe orienté et s ∈S. Le degré sortant (resp. entrant) de u, noté d+(u) (resp. d-(u)) est le nombre d'arcs sortant/entrant dans ce sommet, cad d+(u) = |Γ+(u)|(resp. d-(u) = |Γ+(u)|)

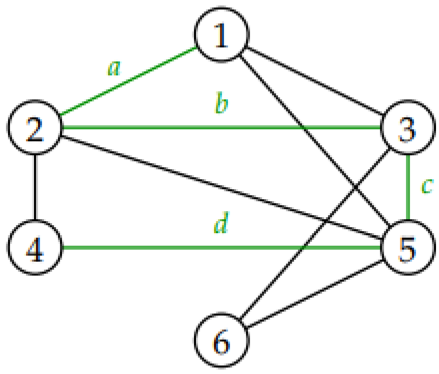


$$d^+(1) = 1 \qquad d^-(1) = 2$$
$$d^+(2) = 3 \qquad d^-(2) = 1$$
$$d^+(3) = 1 \qquad d^-(3) = 2$$
$$d^+(4) = 3 \qquad d^-(4) = 1$$
$$d^+(5) = 0 \qquad d^-(5) = 2$$

- Soit G = (S, A) un graphe non orienté. Une chaîne de u à v, notée µ (u, v), est une suites d'aretes consécutives reliant u à v.
- Une chaine est simple/élémentaire) si elle ne passe pas deux fois par la meme arete/meme sommet.

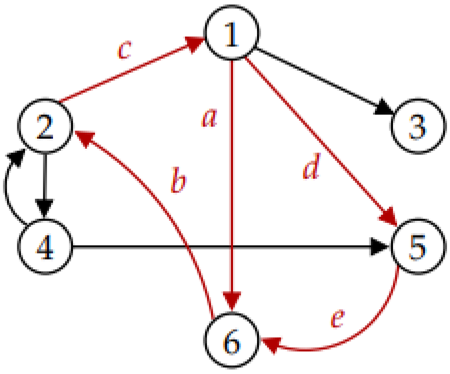


Chaîne

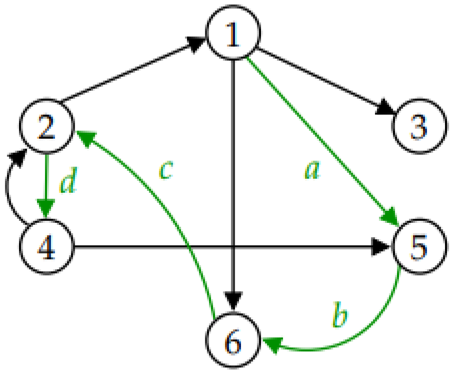


Chaîne simple élémentaire

- Soit G = (S, A) un graphe orienté. Un chemin de u à v, noté µ[u, v], est une suite d'arcs consécutifs reliant u à v
- Un chemin est simple/élémentaire s'il ne passe pas deux fois par le même arc/meme sommet.

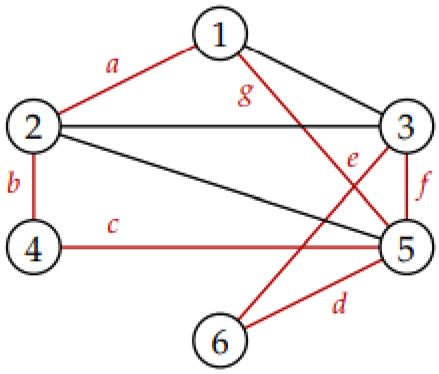


Chemin simple

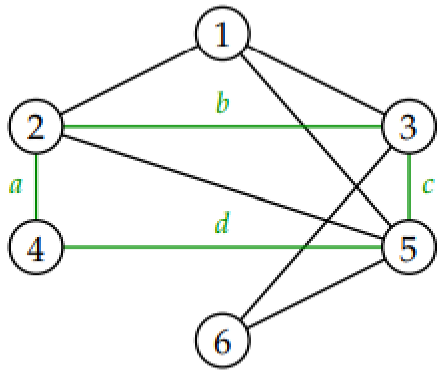


Chemin élémentaire

- Soit G = (S, A) un graphe non orienté. Un cycle de G est une chaine dont les extrémités coïncident
- Un cycle est simple (resp. élémentaire) s'il est une chaîne simple (resp. élémentaire).

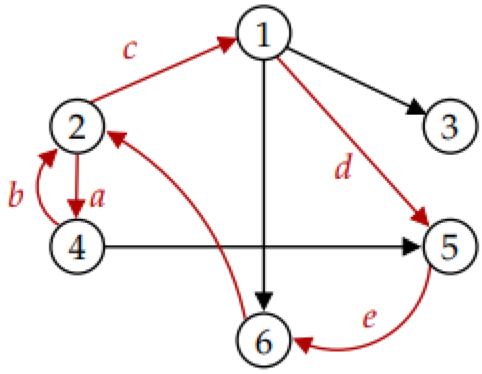


Cycle simple

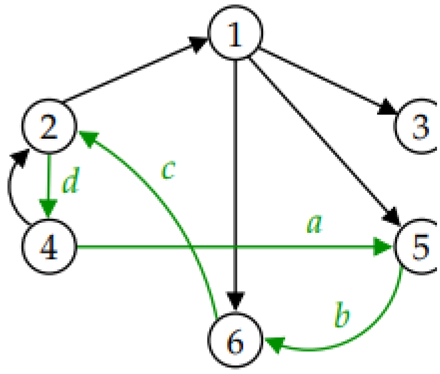


Cycle simple élémentaire

- Soit G = (S, A) un graphe orienté. Un cycle de G est un circuit dont les extremités coïncident
- Un circuit est simple/élémentaire s'il est un chemin simple/élementaire



Circuit simple

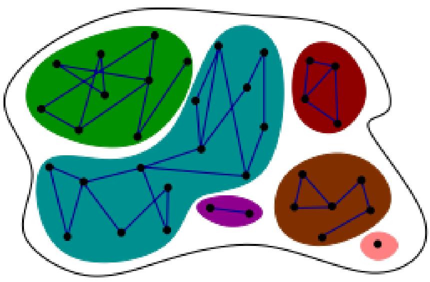


Circuit simple élémentaire

CONNEXITE (FORTE)

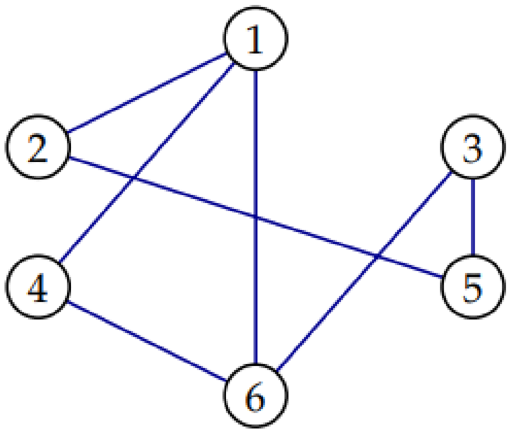
Equivalence :

- Soit E un ensemble et R une relation de E dans E, R est une relation d'équivalence si et seulement si elle est réflexive, symétrique et transitive.
- Soit E un ensemble, x un élément de E, et R une relation d'équivalence de E dans E. La classe d'équivalence [x] est l'ensemble des y ∈ E tels que :
  y ∈ [x] ⇔ xRy

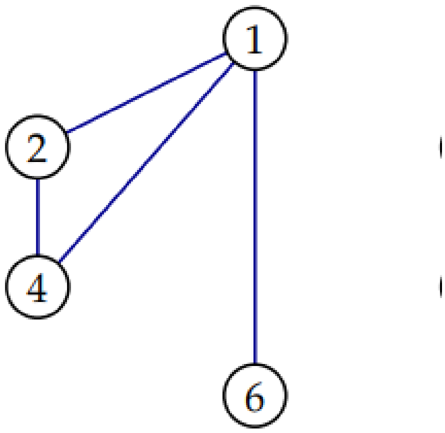


Connexité

- Un graphe G = (S, A) est connexe si, ∀u, v ∈ S, ∃µ(u, v)
- La relation uRv ⇔ (u = v) ou (∃µ(u, v)), est une relation d'equivalence et les classes d'equivalence induites sur S forment une partition {S_1,...,S_p}
- Les sous-graphes C_1,...,C_p engendrés par les sous-ensembles S_1,...,S_p sont les composantes connexes de G.



Graphe connexe

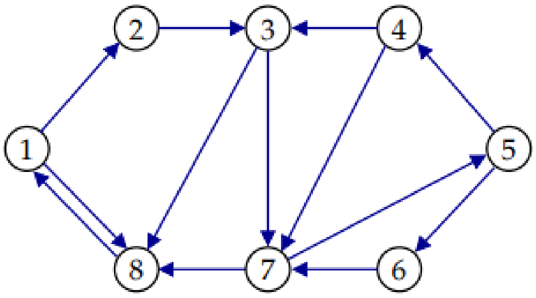


Graphe non connexe

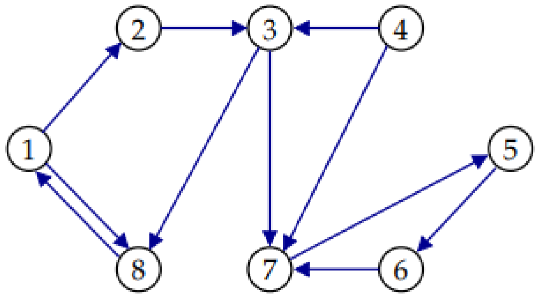
Connexité (forte)

- Un graphe orienté G = (S, A) est fortement connexe si, ∀u, v ∈ S, ∃µ[u, v] et ∃µ[v, u]
- La relation uRv ⇔ (u = v) ou (∃µ[u, v] et ∃µ[v, u]) est une relation d'équivalence et les classes d'équivalence induites sur S forment une partition {S_1,...,S_P}.

- Les sous-graphes C_1,…,C_p engendrés par les sous-ensembles S_1,…,S_p sont les composantes connexes de G.

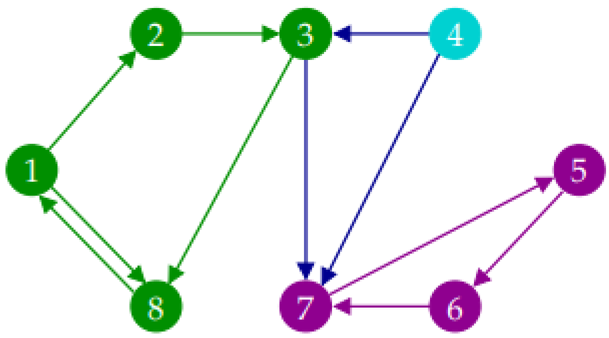


Graphe fortement connexe



Graphe non fortement connexe

Calcul des composantes fortement connexes :



Prédecesseurs	Sommets	Successeurs
{2,3,4,8}	1	{2,3,5,6,7,8}
∅	2	∅
∅	3	∅
∅	4	{5,6,7}
{6,7}	5	{6,7}
∅	6	∅
∅	7	∅
∅	8	∅

- Fortement connexe si ils sont predesseseur et succeseurs d’un sommet -> on n’aurait pas besoin de calculer le sommet de ces nombres car déjà dans une fortement connexe
- Quand predesseseur et successeur d’un sommet sont fortement connexe et déjà calculer dans le tableau on ne les ets pas car déjà calculer.

- Algorithme de Foulkes :

```
Procédure Foulkes(d G = (S, A))
    C[1..|S|], D[1..|S|], M[1..|S|] : tab_ensemble ;
Début
    Pour tout s ∈ S faire
        C[s] := ∅ ;
        D[s] := ∅ ;
        M[s] := ∅ ;
    Fin faire
    Pour tout s ∈ S faire
        Si (C[s] = ∅) alors
            C[s] := {s} ;
            montée(G, s, C, M[s]) ;
            descente(G, s, C, D[s]) ;
            Pour tout (x ∈ D[s] et x ∈ M[s]) faire
                C[s] := C[s] ∪ {x} ;
            Fin faire
            Pour tout (x ∈ C[s]) faire
                C[x] := C[s] ;
            Fin faire
        Fin si
    Fin faire
Fin
```

```
Procédure descente(d G = (S, A), d u ∈ S, d C : tab_ensemble, dr D : ensemble)
Début
    D := D ∪ {u} ;
    Pour tout v t.q. (u, v) ∈ A, v ∉ D et C_v = ∅ faire
        descente(G, v, D) ;
    Fin faire
Fin
```

```
Procédure montée(d G = (S, A), d v ∈ S, d C : tab_ensemble, dr M : ensemble)
Début
    M := M ∪ {v} ;
    Pour tout u t.q. (u, v) ∈ A, u ∉ M et C_u = ∅ faire
        montée(G, u, M) ;
    Fin faire
Fin
```

CHAINES ET CYCLES EULERIENS

Cycles Euleriens

- Theorème : Un graphe non orienté connexe admet un cycle qui traverse chaque arete une et une seule fois ssi chaque sommet a un degré pair. Un tel cycle s'appelle un cycle eulérien.
- Proposition : Soit G un graphe dont tous les sommets sont de degré pair. Alors G peut être divisé en cycles, contenant toutes les arêtes de G, deux quelconques d'entre eux n'ayant aucune arête en commun.

Démonstration :

Soit $G = (S, A)$ un graphe t.q. $\forall s \in S,\ d(s) \equiv 0 \mod 2$. Un premier cycle de G est obtenu en partant d'un sommet i et en parcourant différentes arêtes sans les retraverser. Comme G contient un nombre fini d'arêtes, il existe un moment où l'on arrive sur un sommet s dont on ne peut ressortir qu'en repassant par une arête déjà traversée. Soit a cette arête. L'ensemble des arêtes de la chaîne obtenue entre les deux occurrences de a forment un premier cycle C_1. Construisons un graphe $G' = (S, A' = A \setminus C_1)$.

G' est éventuellement non connexe. En revanche, il est constitué de sommets de degré pair. Si G' contient des arêtes, on peut reproduire avec G' la démarche précédente pour obtenir un cycle C_2 n'ayant aucune arête commune avec C_1.

En reproduisant la démarche jusqu'à ce qu'il ne reste plus d'arête, les cycles $C_1, \ldots, C_k$ obtenus, avec $k \leqslant |A|$, sont tels que $\bigcup_{i=1}^{k} C_i = A$ et que deux quelconques d'entre eux n'ont aucune arête en commun. □

- Théorème : G non orienté connexe admet un cycle eulérien ⇔ ∀s ∈S, d(s) = 0 [2].

Démonstration :

Montrons tout d'abord que la condition est nécessaire. Si G admet un cycle eulérien, alors chaque traversée d'un sommet i utilise 2 arêtes incidentes à i et ajoute 2 au degré de i. Puisque chaque arête est traversée une seule fois, le degré de chaque sommet est une somme de 2, c'est-à-dire un nombre pair.

Montrons maintenant qu'elle est suffisante. Comme tous les sommets de G sont de degré pair, G peut être divisé en cycles recouvrant A qui n'admettent aucune arête en commun.

Considérons l'un de ces cycles C_1. Si C_1 contient toutes les arêtes de G, C_1 est le cycle eulérien de G. Sinon, partant d'un sommet quelconque, on parcourt C_1 jusqu'à rencontrer un sommet appartenant à un second cycle C_2. Ce sommet existe puisque G est connexe. C_2 est parcouru, puis C_1.

Si cette chaîne contient toutes les arêtes de G, on obtient le cycle eulérien de G. Sinon, on parcourt cette chaîne jusqu'à rencontrer un sommet d'un cycle C_3. Puisque G est connexe, ce cycle existe.

En procédant ainsi jusqu'à ce que tous les cycles de G aient été parcourus, on obtient une chaîne dont les deux extrémités sont confondues et telle que chaque arête a été visitée une fois et une seule, ce qui montre que G admet un cycle eulérien. □

(Algorithme de Hierholzer : démonstration sur diapo)

Théorème de la chaine eulérienne :
- Une graphe non orienté connexe admet une chaine qui traverse chaque arete une et une seule fois si et seulement s'il admet zéro ou deux sommets de degré impair. Une telle chaîne s'appelle une chaîne eulérienne.

- G non orienté connexe admet une chaîne eulérienne ⇔0 ou 2 s ∈S, d(s) = 1 [2]

Démonstration :

Soit u et v les deux extrémités de cette chaîne. Si $u = v$ alors la chaîne est un cycle eulérien et il y a donc zéro sommet de degré impair. Considérons à présent le cas où $u \neq v$. En ajoutant une arête entre u et v, on obtient un graphe admettant un cycle eulérien dont tous les sommets sont de degré pair. Si l'on supprime l'arête entre u et v, on retrouve G et u et v sont les seuls sommets de degré impair.

Considérons maintenant un graphe non orienté connexe G admettant zéro ou deux sommets de degré impair. Dans le cas où il en admet zéro, G admet un cycle eulérien et donc une chaîne eulérienne. Dans le cas où il en admet deux, l'ajout d'une arête entre les deux sommets de degré impair u et v permet de construire un graphe admettant un cycle eulérien. En enlevant $\{u, v\}$, on obtient une chaîne contenant une seule fois toutes les arêtes de G. G admet donc une chaîne eulérienne. □

COLORATION

Coloration de cartes.

- Théorème Il est possible de colorier avec quatre couleurs distinctes n'importe quelle carte découpée en régions connexes, de sorte que deux régions ayant une frontière en commun soient de couleurs distinctes.
- On peut voir ses régions comme des graphes. -> colorier les sommets du graphe. = pas 2 sommet voisins avec la même couleur. = graphe planaire

- Def : un graphe planaire non orienté qu'on peut dessiner sur un plan sans qu'aucune arête n'en croise une autre est un graphe planaire.
- Def : Un graphe non orienté G est planaire ssi il existe une carte ayant G comme graphe associé.
- Algorithme de Welsh-Powell:
  - Trier les sommets du graphe par ordre décroissant de leur degré
  - On choisit une couleur (bleu sur l'exemple)
  - Tant qu'il y a encore des sommets en noir faire
  - Parcourir la liste triée de sommets en noir et colorier les sommets qui ne sont pas connectés à d'autres sommets de la couleur en cours
  - Choisir une nouvelle couleur