Rappels de théorie des graphes

Correction du Quizz

2010-2011

Q1,Q2 - Définitions / Notations

Soit $G = (V, E)$ ....

- Notation convenant pour un graphe non orienté/orienté.
- Ensemble des sommets V, ensemble des arêtes/arcs E.
- Nombre de sommets n, nombre d'arêtes/arcs m.
- Notation : $xy \in E$, où xy vaut pour $\{x, y\}$ (non orienté) / pour (x, y) (orienté).
- Par défaut, nos graphes seront sans boucles et sans arêtes/arcs multiples (toujours le préciser au démarrage car il existe pleins de variantes dans la définition des graphes, préciser aussi si vous êtes en non orienté ou en orienté).

Notations à connaître (à peu près standard) : soit $x \in V$,

- Voisinage de x (non orienté) : $N(x) = \{y \in V \mid xy \in E\}$
- Voisinage sortant de x (orienté) : $N^+(x) = \{y \in V \mid xy \in E\}$
- Voisinage entrant de x (orienté) : $N^-(x) = \{y \in V \mid yx \in E\}$
- Degré $d(x) = |N(x)|$, degré sortant $d^+(x) = |N^+(x)|$, degré entrant $d^-(x) = |N^-(x)|$

Définitions à connaître :

- Chemin, cycle (non orienté ou orienté).
- Chemin/cycle simple : ne passe pas deux fois par le même sommet.
- Connexité (non orienté), forte connexité (orienté).
- Sous-graphe, sous-graphe induit, mineur, complémentaire.
- Clique (non orienté), indépendant (non orienté), ...

Notations courantes (non orienté) :

- Chemin simple à n sommets : P_n
- Cycle simple à n sommets : C_n
- Clique à n sommets : K_n
- Indépendant à n sommets : I_n

Graphes particuliers (non orienté) :

- Forêt : sans cycle.
- Arbre : sans cycle et connexe.
- Biparti : ensemble de sommets partitionnable en deux ensembles indépendants.
- Biparti complet $K_{n,p}$: toutes les arêtes entre un I_n et un I_p.
- Régulier : tous les sommets de même degré.

Q3 - Indépendants, cliques, couvertures

Il existe des graphes où S et $V \setminus S$ sont des indépendants (prendre par exemple le graphe N et prendre le bas du N comme indépendant S).

⚠ dans l'énoncé, on ne considère que le complémentaire $\overline{S} = V \setminus S$ de S et pas le complémentaire $\overline{G} = (V, \overline{E})$ de G dont les arêtes sont $\overline{E} = \{xy \mid x, y \in V,\ xy \notin E\}$.

Théorème (folklore)

Soit $G = (V, E)$ non orienté et $S \subseteq V$, on a l'équivalence :

- *S indépendant dans G.*
- *S clique dans $\overline{G}$.*
- *$\overline{S}$ couvre toutes les arêtes de G.*

Q4,Q5 - Un petit théorème classique sur les degrés

Reformulation des questions Q4,Q5 :

- Hypothèse “St Valentin” :

  envoi réciproque de carte $\Rightarrow$ graphe non orienté.
- Q4 : existe-t-il un graphe non orienté 2-régulier à 9 sommets ?
- Q5 : existe-t-il un graphe non orienté 3-régulier à 9 sommets ?

Reformulation des questions Q4,Q5 :

- Hypothèse “St Valentin” :

  envoi réciproque de carte $\Rightarrow$ graphe non orienté.
- Q4 : existe-t-il un graphe non orienté 2-régulier à 9 sommets ?
- Q5 : existe-t-il un graphe non orienté 3-régulier à 9 sommets ?

Théorème (folklore)

Soit $G = (V, E)$ un graphe non orienté. Alors $\sum_{x \in V} d(x) = 2|E|$.

Corollaire

Soit $G = (V, E)$ un graphe non orienté. Alors le nombre de sommets de degré impair est pair.

Q6 - Graphes eulériens

Definition

Cycle/chemin eulérien : passe par toutes les arêtes/arcs une fois et une seule.

Théorème (Euler 1736)

G non orienté admet un cycle eulérien ssi il est connexe et tous ses sommets sont de degré pair.

Théorème (Euler 1736)

G non orienté admet un chemin eulérien ssi il est connexe et le nombre de sommets de degré impair est 0 ou 2.

Théorème (folklore)

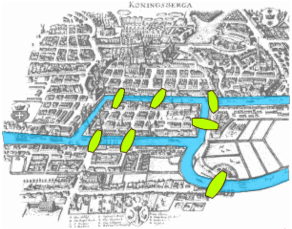
G orienté admet un cycle eulérien ssi il est fortement connexe et pour tout sommet x, on a $d^-(x) = d^+(x)$.

Ponts de Venise : eulérien ?

Q6 - Graphes eulériens

Ponts de Venise : eulérien ?

Ponts de Königsberg (version barque) : est-il possible de passer une fois et une seule sous chaque pont ?



Q7 - Sous-graphe interdit

Théorème (folklore)

Soit G un graphe non orienté connexe. Si G n'est pas une clique, alors il contient un sous-graphe induit *isomorphe à P_3.*

La réciproque est clairement vraie, on a donc une caractérisation des cliques par sous-graphe induit interdit : G clique ssi G connexe sans P_3 induit (P_3-free).

- G sans P_2 induit ?

Q7 - Sous-graphe interdit

Théorème (folklore)

Soit G un graphe non orienté connexe. Si G n'est pas une clique, alors il contient un sous-graphe induit *isomorphe à P_3.*

La réciproque est clairement vraie, on a donc une caractérisation des cliques par sous-graphe induit interdit : G clique ssi G connexe sans P_3 induit (P_3-free).

- G P_2-free $\iff$ G isomorphe à I_n

Q7 - Sous-graphe interdit

Théorème (folklore)

Soit G un graphe non orienté connexe. Si G n'est pas une clique, alors il contient un sous-graphe induit *isomorphe à P_3.*

La réciproque est clairement vraie, on a donc une caractérisation des cliques par sous-graphe induit interdit : G clique ssi G connexe sans P_3 induit (P_3-free).

- G P_2-free $\iff$ G isomorphe à I_n
- G sans P_3 induit ?

Théorème (folklore)

Soit G un graphe non orienté connexe. Si G n'est pas une clique, alors il contient un sous-graphe induit *isomorphe à P_3.*

La réciproque est clairement vraie, on a donc une caractérisation des cliques par sous-graphe induit interdit : G clique ssi G connexe sans P_3 induit (P_3-free).

- G P_2-free $\iff$ G isomorphe à I_n
- G P_3-free $\iff$ G union disjointe de cliques

Q7 - Sous-graphe interdit

Théorème (folklore)

Soit G un graphe non orienté connexe. Si G n'est pas une clique, alors il contient un sous-graphe induit *isomorphe à P_3.*

La réciproque est clairement vraie, on a donc une caractérisation des cliques par sous-graphe induit interdit : G clique ssi G connexe sans P_3 induit (P_3-free).

- G P_2-free $\iff$ G isomorphe à I_n
- G P_3-free $\iff$ G union disjointe de cliques
- G sans P_4 induit ?

Q7 - Sous-graphe interdit

Théorème (folklore)

Soit G un graphe non orienté connexe. Si G n'est pas une clique, alors il contient un sous-graphe induit *isomorphe à P_3.*

La réciproque est clairement vraie, on a donc une caractérisation des cliques par sous-graphe induit interdit : G clique ssi G connexe sans P_3 induit (P_3-free).

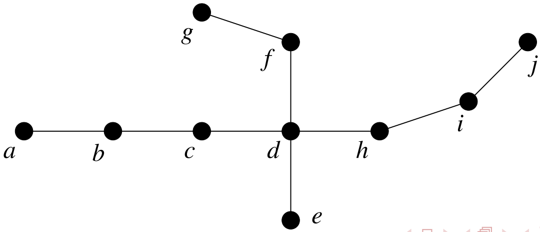
- G P_2-free $\iff$ G isomorphe à I_n
- G P_3-free $\iff$ G union disjointe de cliques
- G P_4-free $\iff$ G cographe

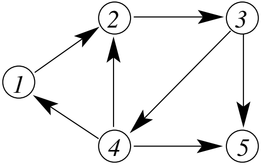
Définition

- *Couplage dans un graphe non orienté $G = (V, E)$: ensemble d'arêtes $M \subseteq E$ qui n'ont pas d'extrémités en commun.*
- *Couplage parfait : couplage couvrant tous les sommets (càd tout sommet est l'extremité d'une des arêtes du couplage).*

Lemme

Si G graphe non orienté admet un couplage parfait, alors son nombre de sommets est pair.

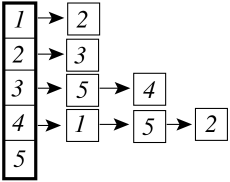




Graphe oriente

$$\begin{array}{c|ccccc|} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & 1 & 0 & 0 & 0 \\ 2 & 0 & 0 & 1 & 0 & 0 \\ 3 & 0 & 0 & 0 & 1 & 1 \\ 4 & 1 & 1 & 0 & 0 & 1 \\ 5 & 0 & 0 & 0 & 0 & 0 \\ \hline \end{array}$$

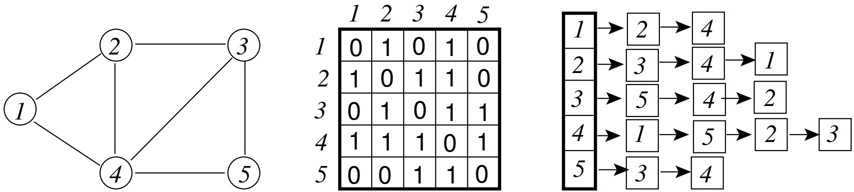
Matrice d'adjacence



Listes d'adjacence

	Matrice d'adj.	Listes d'adj. $N^+(x)$
Espace de stockage	$\Theta(n^2)$	$\Theta(n+m)$
Test d'adjacence $xy \in E$	$O(1)$	$O(d^+(x))$

Q9 - Structures de données classiques (non orienté)



Graphe non oriente | Matrice d'adjacence | Listes d'adjacence

	Matrice d'adj.	Listes d'adj. $N(x)$
Espace de stockage	$\Theta(n^2)$	$\Theta(n+m)$
Test d'adjacence $xy \in E$	$O(1)$	$O(d(x))$

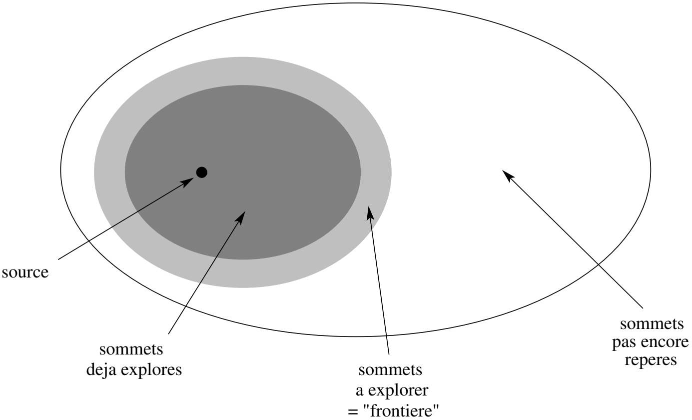
Q9 - Parcours de graphe : principe (orienté / non orienté)

Algorithmes de graphes : algorithmes "myopes" à la base, car souvent informations locales en entrée (p.ex. voisinages) qu'il faut collecter et organiser pour résoudre des problèmes demandant une vue globale.

Prétraitement classique : commencer par explorer tout le graphe avec un/des *parcours*.

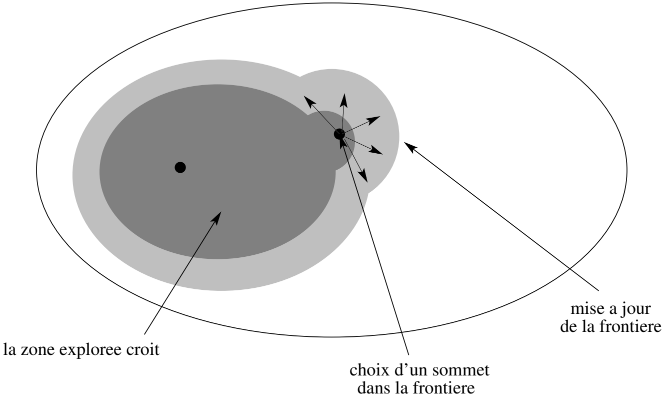
Ingrédients d'un parcours : point de départ (source) unique, un peu de mémoire en plus (marquage des sommets pour éviter de boucler, stockage des sommets repérés mais pas encore explorés), construction d'un arbre de parcours orienté et enraciné.

Q9 - Parcours de graphe : principe (orienté / non orienté)



Instant t

Q9 - Parcours de graphe : principe (orienté / non orienté)



Instant t+1

Parcours classiques :

- Parcours en largeur (BFS = Breadth First Search).
- Parcours en profondeur (DFS = Depth First Search).

ANIMATION CatBox

Complexité : dans les deux cas, implémentable en linéaire en la taille de l'entrée, càd $O(n+m)$ pour les listes d'adjacence, $O(n^2)$ pour la matrice d'adjacence.

Connexité (non orienté) : notion globale.

Algorithme

- *Choisir un sommet arbitraire.*
- *Faire un parcours quelconque depuis ce sommet et vérifier que le nombre de sommets explorés est bien n.*

Complexité : linéaire.

Forte connexité (orienté) : il existe des algorithmes simples et linéaires en 2 parcours successifs.

Q10 - Recherche de cycle (cas orienté)

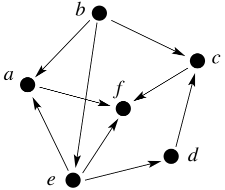
Théorème (folklore)

G orienté est sans cycle (DAG : directed acyclic graph) ssi G admet un tri topologique, càd une numérotation des sommets $\pi : V \to \{1, \ldots, n\}$ telle que si $xy \in E$ alors $\pi(x) < \pi(y)$.

Algorithme (tri topo = séquence d'élimination des sommets)

- *tant que $\exists x,\ d^-(x) = 0$, faire $G \leftarrow G \setminus x$;*
- *si G vide, alors "G est un DAG", sinon "G admet un cycle".*

Complexité : implémentable en linéaire, càd $O(n+m)$ ou $O(n^2)$.



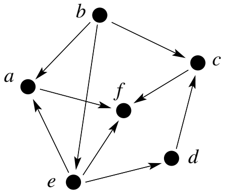
Théorème (folklore)

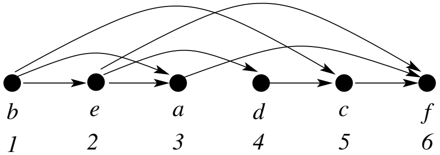
G orienté est sans cycle (DAG : directed acyclic graph) ssi G admet un tri topologique, càd une numérotation des sommets $\pi : V \to \{1,\ldots,n\}$ telle que si $xy \in E$ alors $\pi(x) < \pi(y)$.

Algorithme (tri topo = séquence d'élimination des sommets)

- *tant que $\exists x,\ d^-(x) = 0$, faire $G \leftarrow G \setminus x$;*
- *si G vide, alors "G est un DAG", sinon "G admet un cycle".*

Complexité : implémentable en linéaire, càd $O(n+m)$ ou $O(n^2)$.





Q11 - Recherche de cycle (cas non orienté)

⚠ Ne pas appliquer bètement la Q10 en regardant un graphe non orienté comme un graphe orienté via la transformation classique qui interprète la présence de l'arête $\{x, y\}$ comme la présence des deux arcs (x, y) et (y, x). Ici cela crée des cycles de taille 2 impossibles à enlever par tri topologique.

Parcours DFS ou BFS marchent ici : si une arête qui ramène sur un sommet déjà découvert et qui n'est pas le père dans l'arbre de parcours (càd sans ré-emprunter une arête déjà vue), alors il existe un cycle $\Rightarrow$ complexité en $O(n+m)$

Amélioration de l'analyse de complexité : sans cycle = forêt $\Rightarrow$ $m \leq n-1$, donc en faisant le parcours ci-dessus, si le nombre d'arêtes empruntées dépasse strictement $n-1$, c'est qu'il existe un cycle avec ces arêtes qui est forcément répéré par le parcours $\Rightarrow$ complexité en $O(n)$.

Définition

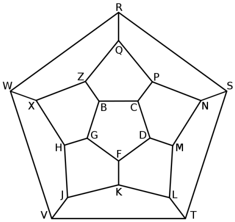
Cycle hamiltonien : cycle passant une fois et une seule par chaque sommet.

Théorème (Karp 1972)

Décider si un graphe G contient un cycle hamiltonien est NP-complet (valable en orienté et en non orienté).

⚠ Ne pas confondre avec cycle eulérien.

"The Icosian Game" inventé par Sir William Hamilton (1857)



Algorithme (Jarnik 1930 / Prim 1957)

- *Parcours depuis une source, en assignant un poids à chaque sommet de la frontière = poids de la plus petite arête vers la zone déjà explorée.*
- *Progression en choisissant un sommet de poids min dans la frontière.*
- *Structure de données pour gérer la frontière : file de priorité (priority queue).*

ANIMATION CatBox

File de priorité	Tableau	Tas binaire	Tas de Fibonnaci
Complexité	$O(n^2)$	$O(m \log n)$	$O(m + n \log n)$

Algorithme (Kruskal 1956)

- *Trier les arêtes selon leur poids.*
- *Ajouter les arêtes une à une par poids croissant, en éliminant les arêtes qui créent un cycle avec celles précédemment ajoutées.*
- *Structure de données pour tester la création de cycle : union-find.*

ANIMATION CatBox

Union-Find	Listes	Listes pondérées	Arbres de Tarjan
Complexité (sans le tri des arêtes)	$O(mn)$	$O(m \log n)$	$O(m\alpha(m,n))$

Poids homogènes : si les poids des arêtes valent tous 1, alors tous les arbres couvrants sont optimaux car ils ont tous le même poids, à savoir $n-1$.

Algorithme : prendre l'arbre d'un parcours quelconque (p.ex. DFS ou BFS) $\Rightarrow$ complexité linéaire.

Q15,Q16 - Calcul de distances : graphe orienté, source unique, poids positifs

Algorithme (Dijskstra 1959)

- *Parcours depuis une source, en assignant un poids à chaque sommet de la frontière = poids du plus court chemin depuis la source à travers la zone déjà explorée.*
- *Progression en choisissant un sommet de poids min dans la frontière.*
- *Structure de données pour gérer la frontière : file de priorité (priority queue).*

ANIMATION CatBox

Gestion de la frontière	Tableau	Tas binaire	Tas Fibonacci
Complexité totale	$O(n^2)$	$O(m \log n)$	$O(m + n \log n)$

Q15,Q16 - Calcul de distances : graphe orienté, source unique, poids positifs

Variantes du problème :

- Cas non orienté : remplacer arête $\{x, y\}$ de poids w_{xy} par deux arcs (x, y) et (y, x) de même poids.
- Poids positifs tous identiques : parcours BFS calcule les distances (en nombre d'arcs/arêtes) depuis la source.
- Présence de poids négatifs : Dijkstra échoue, utiliser d'autres algorithmes comme Bellman-Ford.

Q17 - Calcul de distances : graphe orienté, tous couples, poids quelconques mais sans circuit négatif

Algorithme (Roy-Floyd-Warshall 1962)

- *Numéroter arbitrairement les sommets de* $G = (V, E)$*, ce qui revient à* $V = \{1, \ldots, n\}$*, chaque arc ij ayant pour poids* w_{ij}*.*
- *Introduire les variables* $d_{ij}^{(k)}$ $=$ *plus court chemin de i à j utilisant des sommets intermédiaires dont les numéros sont dans* $\{1, \ldots, k\}$*.*
- *Trouver une formule de récurrence pour calculer ces variables (celles qui nous intéressent à la fin sont les* d_{ij}^{n}*) :*

$$d_{ij}^{(k)} = \begin{cases} w_{ij} & \textit{si } k = 0 \\ \min(d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{(k-1)}) & \textit{si } k \geq 1 \end{cases}$$

Programmation dynamique de complexité $O(n^3)$.
Amélioration par Johnson (1977) mixant Dijsktra + Bellman-Ford + Tas de Fibonacci → complexité en $O(n^2 \log n + mn)$.

Q18 - Reconnaissance des graphes bipartis

Théorème (folklore)

Soit $G = (V, E)$ graphe non orienté, alors on a l'équivalence :

- *G biparti,*
- *G 2-colorable,*
- *G n'admet pas de cycle impair.*

Reconnaissance : faire un parcours (p.ex. DFS ou BFS) et colorer les sommets au fur et à mesure tant qu'on ne rencontrer pas d'arête reliant des sommets de même couleur.

Correction : si terminaison sans problème, alors on dispose d'une 2-coloration. Si un problème est survenu, on peut exhiber un cycle impair (et donc le graphe n'est pas biparti).

Complexité : linéaire.

⚠ Petit piège : faire attention à quels paramètres sont fixés et quelles sont les entrées du problème.

- Q19 : NP-complet (Karp 1972).
- Q20 : P car il suffit de tester les $\binom{n}{k}$ cliques potentielles, soit un algorithme en $O(n^k)$ où k est fixé.
- Q21,Q22 : dès $k = 3$, tester la 3-colorabilité est NP-complet (Karp 1972).

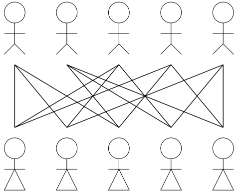
Reformulation de la question : existe-t-il un couplage parfait dans tout biparti 3-régulier avec 5 sommets de chaque coté ?

Théorème (Hall 1935)

Soit $G = (X \cup Y, E)$ un graphe biparti où X et Y sont des indépendants, pour tout $A \subseteq X$, on note $N(A) = \{y \in Y \mid \exists x \in A,\ xy \in E\}$. Alors G admet un couplage parfait ssi pour tout $A \subseteq X$, $|N(A)| \geq |A|$.

Corollaire

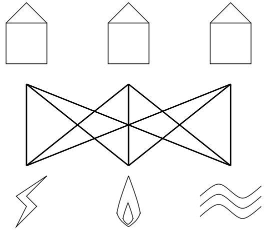
Tout graphe biparti k-régulier admet un couplage parfait.



Reformulation de la question : est-ce que le graphe biparti $K_{3,3}$ est planaire ?

Théorème (Kuratowski 1930)

G non orienté est planaire ssi il n'admet ni K_5, ni $K_{3,3}$ comme mineur.



Q25 - Coloration de graphes

Reformulation de la question : le graphe de l'amitié qui est non orienté et tel que tous les sommets sont de degré ≤ 4, est-il toujours 5-coloriable (1 couleur = 1 groupe) ?

Théorème (folklore)

Soit G un graphe non orienté, alors $\chi(G) \leq \Delta(G) + 1$ avec :

- *$\chi(G)$: nombre minimum de couleurs permettant de colorer les sommets sans avoir deux sommets adjacents de même couleur (nombre chromatique de G).*
- *$\Delta(G)$: degré maximum des sommets dans G.*

Algorithme (coloriage glouton)

choisir un ordre arbitraire de coloriage des sommets et colorier à chaque étape avec une couleur dans $\{1, \ldots, \Delta(G) + 1\}$ pas utilisée parmi les voisins déjà coloriés $\rightarrow$ toujours possible.

Il existe un graphe G tel que $\Delta(G) = 5$ et $\chi(G) = 5$: la clique K_5 !

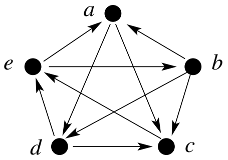
Il existe un graphe G tel que $\Delta(G) = 5$ et $\chi(G) = 5$: la clique K_5 !

Théorème (Brooks 1941)

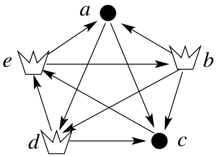
Si G est un graphe connexe différent de C_{2p+1} et K_p, alors $\chi(G) \leq \Delta(G)$.

Q27 - Rois dans les tournois

Reformulation de la question : relation “être plus fort” génère un graphe orienté où entre deux sommets il y a toujours un et un seul arc → ces graphes orientés sont appelés des *tournois*.



Reformulation de la question : relation “être plus fort” génère un graphe orienté où entre deux sommets il y a toujours un et un seul arc → ces graphes orientés sont appelés des *tournois*.



Théorème (folklore)

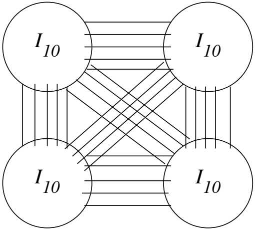
Soit $G = (V, E)$ un tournoi et $x \in V$. Alors x est un Roi ssi $N^+(x)$ est maximal pour $\subseteq$.

Corollaire

Dans tout tournoi, il existe au moins un Roi : par exemple tous les sommets x dont le cardinal $|N^+(x)|$ est maximum.

Q28 - Graphes extrémaux

Maximum d'arêtes pour un graphe à 40 sommets sans clique de taille 5 ? Une construction à 600 arêtes en prenant 4 indépendants I_{10} et en les reliant par toutes les arêtes possibles hors de ces indépendants, soit $\binom{4}{2} \times 10 \times 10$ arêtes.



Pour montrer que c'est le max, utiliser le théorème suivant :

Théorème (Turan 1941)

Si G n'a pas de p-clique, alors $m \leq (1 - \frac{1}{p-1})\frac{n^2}{2}$.

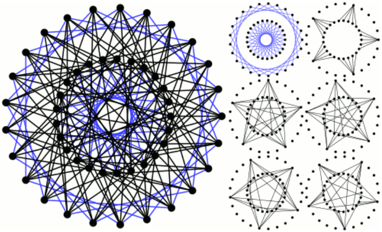
Q29 - Graphes extrémaux

Théorème

Il existe un unique graphe sans triangle, ni carré, de diamètre 2, avec un sommet de degré 7 et **non régulier** *: c'est l'étoile* $K_{1,7}$.

Théorème

Il existe un unique graphe sans triangle, ni carré, de diamètre 2, avec un sommet de degré 7 et **régulier** *: c'est le graphe de Hoffman-Singleton (avec ses 50 sommets et 175 arêtes).*



Fin

Pour plus de précisions et d'autres rappels, n'hésitez pas à ouvrir :

- **"Introduction à l'algorithmique"** de Cormen, Leiserson, Rivest, Stein (pour l'algo).
- **"Introduction to graph theory"** de West (pour les maths).

2 mois en 2 heures : relisez tranquillement le QCM, les réponses, les slides et posez des questions si besoin.

Propositions de sujets bientôt accessibles sur :

`http ://perso.ens-lyon.fr/eric.thierry/Graphes2010`