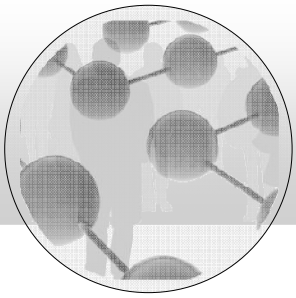
Théorie des graphes



Parcours de graphes et graphes particulières

par

Andréa C. Santos Duhamel

andrea@isima.fr

ISIMA – B108

année 2009/2010

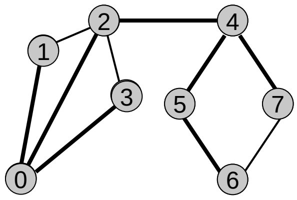
Parcours de graphe

- Visité chaque sommet d'un graphe donné.
- Algorithmes de base qui sont utilisés comme partie de la résolution (ou pour résoudre) plusieurs problèmes importants.
- Plusieurs utilités :
  - Déplacement dans un graphe
  - Problèmes de cheminement
  - Découvert de voisins
  - Vérification de la connexité
  - …

Parcours de graphe

Recherche en largeur

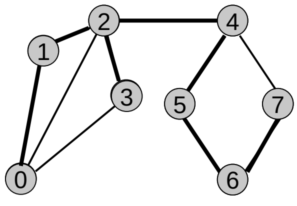
- Structure de donnée du graphe est une liste d'adjacence
- Commencer par un sommet quelconque
- Idée : visiter tous les sommets adjacents du sommet initial, puis avancer dans le graphe.



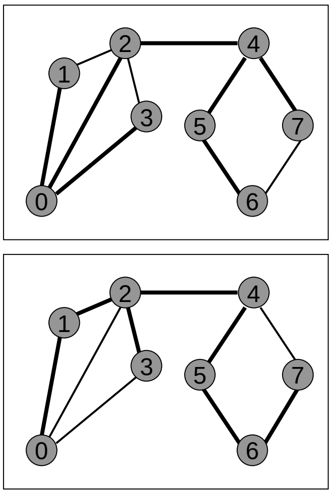
Parcours de graphe

Recherche en profondeur

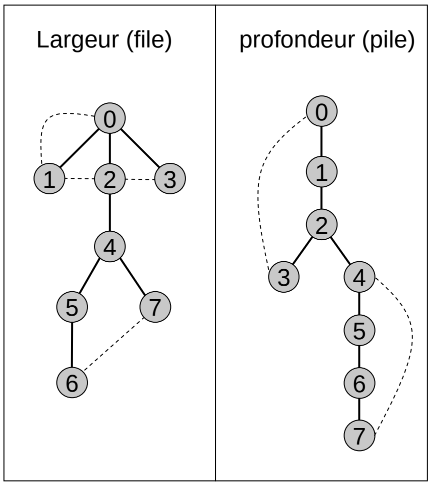
- Structure de donnée du graphe est une liste d'adjacence
- Commencer par un sommet quelconque
- Idée: descendre dans le graphe à partir du sommet initial jusqu'au niveau le plus profond...



Parcours de graphe



Arbres de recherches



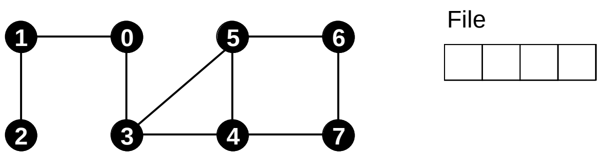
Structures de données file et pile

- Ces deux structures sont très liées car ce sont deux possibilités étendues des listes, mais avec chacune une subtilité.
- Dans une file, la première donnée sortie de cette structure sera la première entrée.
- Dans une pile, la première donnée sortie de cette structure sera la dernière entrée.

Parcours de graphe

Exemple de la logique par le T. Cormen (recherche en largeur)

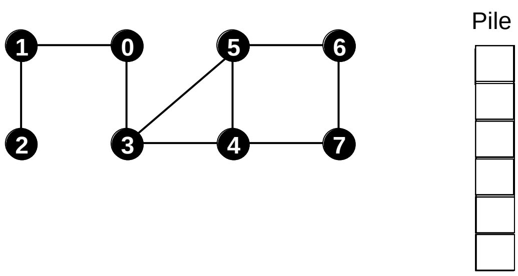
- ○ Sommets non découverts
- ● Sommets découverts (gris)
- ● Tous les sommets adjacentes ont été examinés



Parcours de graphe

Exemple de la logique par le T. Cormen (recherche en profondeur)

- ○ Sommets non découverts
- ● Sommets découverts (gris)
- ● Tous les sommets adjacentes ont été examinés



Algorithme de recherche en largeur

Pseudo-code recherche_largeur(i)

sommet_parcouru(i) ← oui
insérer i dans la file
tant que (file n'est pas vide**) faire**
    j ← prendre le premier sommet de la file
    pour chaque (sommet k adjacent à j) **faire**
        si (sommet_parcouru(k) = non**) alors**
            sommet_parcouru(k) ← oui
            insérer k à la fin de la file

Fonctions récursives

- En informatique les fonctions récursives sont des fonctions dont le calcul nécessite d'invoquer elle-même.
- Un algorithme est dit récursif s'il s'appelle lui-même.
- Exemple du calcul de la factorielle:

n! = n × (n-1) × … × 2 × 1

6! = 6 × 5 × 4 × 3 × 2 × 1 = 720

Pseudo-code factoriel (entier n)

Si (n = 0) **faire**
    retourner 1;
Sinon
    retourner n * factoriel(n - 1);

Algorithme de recherche en profondeur

Pseudo-code recherche_profondeur

pour chaque (sommet i = 1,...,n**) faire**
sommet_parcouru(i) ← non
pour chaque (sommet i = 1,...,n**) faire**
Si (sommet_parcouru(i)) = non**) alors**
empiler(i)

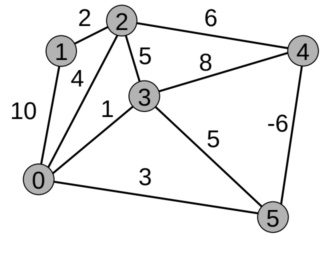
Pseudo-code empiler (i)

sommet_parcouru(i) ← oui
pour chaque (sommet j adjacent à i**) faire**
Si (sommet_parcouru(j) = non**) faire**
empiler (j)

Graphes non-orientés et pondérés

- Il existe une fonction de coût (poids) associés à chaque arêtes du graphe

$\mathbf{C_{ij} \in \Re \ \forall \ [j,i] \in G}$
V = {0,1,2,3,4,5} → |V|=n →n=6
E = {[0,1], [0,2], [0,3], [0,5], [1,2], [2,3], [2,4], [3,4], [3,5], [4,5]} → |E|=m →m=10



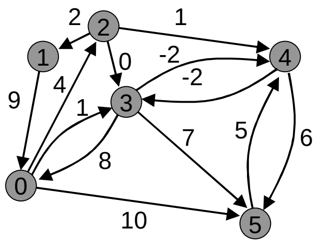
Graphes orientés et pondérés

- Il existe une fonction de coût (poids) associés à chaque arêtes du graphe

$C_{ij} \in \Re \; \forall \; (j,i) \in G$

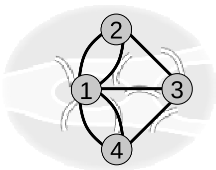
$V = \{0, 1, 2, 3, 4, 5\} \rightarrow |V|=n \rightarrow n=6$

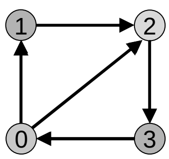
$A = \{(0,2), (0,3), (0,5), (1,0), (1,1), (2,1), (2,3), (2,4), (3,0), (3,4), (3,5), (4,3), (4,4), (4,5), (5,4)\} \rightarrow |A|=m \rightarrow m=15$



Définitions : chemin Eulérien

- Chemin Eulérien : passe une seule fois par chaque arc du graphe
- Un cycle Eulérien est un chemin Eulérien dont l'origine est aussi la destination
- Le célèbre problème du **postier Chinois…**





Chemin Eulérien
Sommet origine = 0
Sommet destination = 2

Il n'y a pas de cycle (circuit) Eulérien dans ces graphes

Un graphe connexe admet un cycle (circuit) eulérien si et seulement si tous ses sommets sont de degré pair.

Le problème du postier chinois

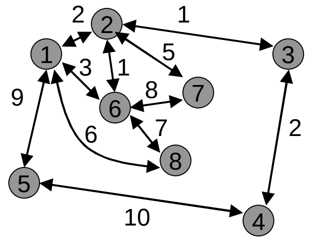
- Problème posé pour la première fois par un chinois (Mei-ko Kwan) pour optimiser la distribution du courrier dans une ville.

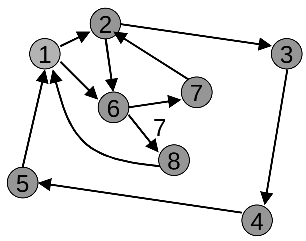
- Le problème du *postier chinois* est de parcourir les rues d'une ville en passant au moins une fois dans chaque rue toute en minimiser la distance totale parcourue.

- Remarques:
  - S'il existe un circuit eulérien le problème est résolu : la somme des valeurs de tous les arcs du circuit est la plus petite valeur possible.
  - Si le graphe n'est pas eulérien, le circuit emprunte nécessairement plusieurs fois certains arcs.

Exemple : le postier chinois

Quel est le degré total du sommet 2 ?

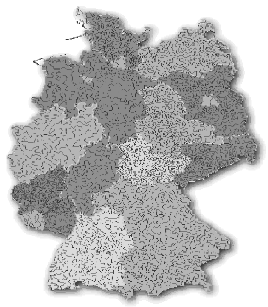




Coût = 54 km

Définitions : chemin Hamiltonien

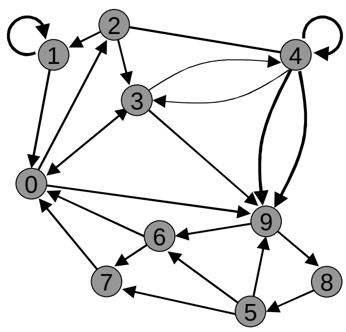
- Chemin Hamiltonien : passe une seule fois par chaque sommet du graphe
- Un cycle (circuit) Hamiltonien est un chemin Hamiltonien dont l'origine est aussi la destination
- Le célèbre problème du **voyageur de commerce…**



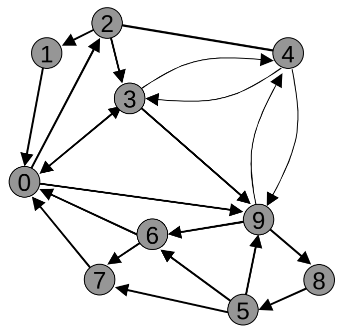
On ne connait pas un algorithme polynomial pour résoudre ce problème.

Graphes particuliers : multiple et simple

Multiple



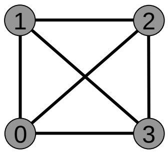
Simple

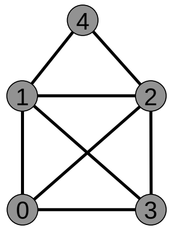


sans boucle,
ni arcs multiples entre les sommets

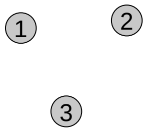
Graphes particuliers : complet et vide

- Graphe complet : si tous les sommets sont adjacents





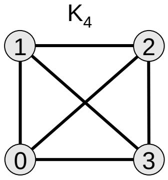
- Graphe vide : si le graphe ne contient aucun arc (arête)



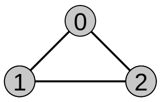
Graphes particuliers : K_p et K-régulier

- K_p est un graphe complet avec p=n sommets et (cf. plus loin) arêtes

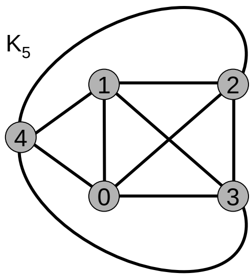
K_4



K_3



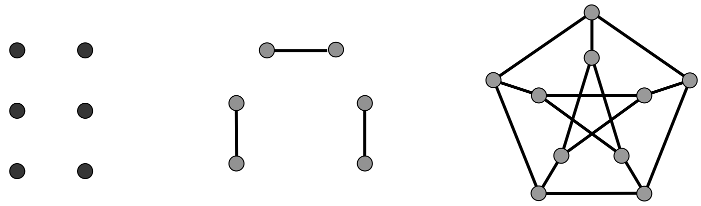
K_5



- Un graphe est *k*-régulier si chacun de ses sommets est de degré *k*; Tous graphe complet K_p est (n-1)-régulier.

Graphes particuliers : K_p et K-régulier

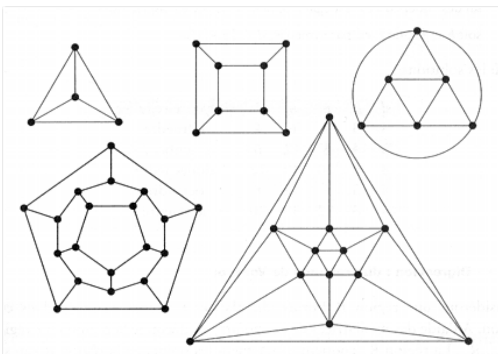
- Exemple de graphes non-complets, mais réguliers



0-régulier 1-régulier 3-régulier
Graphe de Petersen

Graphes particuliers : planaire

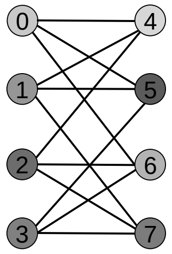
- Graphe planaire : peut être dessiné sans que ses arêtes (arcs) ne se croisent.
- Intérêt : circuit électriques (impression des circuits sur une carte)



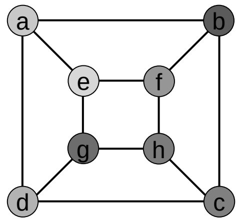
Graphes particuliers : isomorphes

- Si deux graphes sont isomorphes, alors ils ont la même structure : peu importe la façon dont ils sont dessinés ou étiquetés
- Il est possible de déplacer les sommets ou de changer les étiquettes pour que l'un soit la copie conforme de l'autre

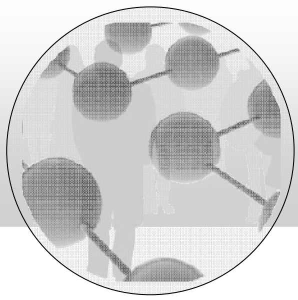
G_1



G_2



Théorie des graphes



Exercises

année 2009/2010