Corrigé

Examen de Théorie des Graphes

EPITA ING1 2013 S2; A. DURET-LUTZ

Durée : 1 heure 30

28 mars 2010

Consignes

- Cet examen se déroule **sans document** et **sans calculatrice**.
- Répondez sur le sujet dans les cadres prévus à cet effet.
- Il y a 6 pages d'énoncé. **Rappelez votre nom en haut de chaque feuille** au cas où elles se mélangeraient.
- Ne donnez pas trop de détails. Lorsqu'on vous demande des algorithmes, on se moque des points-virgules de fin de commande etc. Écrivez simplement et lisiblement. Des spécifications claires et implémentables sont préférées à du code C ou C++ verbeux.
- Pour fêter l'anniversaire de Lady Gaga, le barème (indicatif) correspond à une note sur 25.

1 Complexités diverses (8 points)

Soit $G = (V, E)$ un graphe connexe, non-orienté, et non vide. Supposez que ce graphe vous est fourni sous forme de matrice ou de liste d'adjacence (à votre convenance). Lorsqu'on vous demande la complexité d'un algorithme, exprimez-la en fonction de $|V|$ et $|E|$, en utilisant la notation asymptotique (Θ, O, ou Ω) la plus précise.

1. **(4 pts)** Décrivez un algorithme permettant de calculer à la fois le rayon et le diamètre de G. Quelle est sa complexité ?

Réponse :

Voici une première façon possible. À partir de la matrice d'adjacence, on applique Floyd-Warshall pour calculer une carte des distances entre toutes les paires de sommets en $\Theta(|V|^3)$ opérations. Cette carte des distance est une matrice de taille $|V| \times |V|$. Le minimum de chaque ligne nous donne l'excentricité de chaque sommet. À partir de la matrice, le vecteur des excentricité peut donc se calculer en $\Theta(|V|^2)$ opérations. Le minimum et le maximum des composantes de ce vecteur correspondent respectivement aux rayon et diamètre de G, ils se calculent en $\Theta(|V|)$ opérations.
Nous avons donc

$$\underbrace{\Theta(|V|^3)}_{\text{Floyd-Warshal}} + \underbrace{\Theta(|V|^2)}_{\text{excentricités}} + \underbrace{\Theta(|V|)}_{\text{min et max}} = \Theta(|V|^3)$$

Une seconde approche maintenant. Peut-être plus intuitive, en tout cas plus rapide. On peut calculer l'excentricité d'un sommet avec un parcours en largeur ($\Theta(|E| + |V|)$) si le graphe n'est pas pondéré (ce n'était pas précisé), ou avec un appel de Dijkstra ($(|E| + |V|) \log |V|$ avec l'implémentation du cours) si le graphe est pondéré ; dans les deux cas l'excentricité est la plus grande distance trouvée. Pour le rayon et le diamètre, on répète cet algorithme depuis les $|V|$ sommets. Cela nous donne donc une complexité de $\Theta(|V| \times (|E| + |V|))$ si le graphe n'est pas pondéré, ou $\Theta(|V| \times (|E| + |V|) \log |V|)$ sinon. Naturellement, ces deux complexités se simplifient parce qu'on sait que dans un graphe connexe, $|V| = O(|E|)$: on a donc respectivement $\Theta(|V| \times |E|)$ ou $\Theta(|V| \times |E| \log |V|)$.

2. On considère l'algorithme suivant dans lequel les sommets V sont numérotés de 1 à $|V|$.

```
1.  AlgoSansNom(G = (V, E))
2.    ∀v ∈ V, P[v] ← 0
3.    v0 ← any vertex of V
4.    todo.push_back(v0)
5.    P[v0] ← v0
6.    while (todo ≠ ∅)
7.      v ← todo.pop_front()
8.      for x ∈ NeighboursOf(v) do
9.        if x ≠ P[v] then
10.         if P[x] ≠ 0 then
11.           return false
12.         else
13.           todo.push_back(x)
14.           P[x] ← v
15.   return true
```

(2 pts) Que calcule cet algorithme ? (Inutile de vous justifier.)

Réponse :

Les sommets ajoutés à la file *todo* ne peuvent y être ajouté qu'une fois, car ils sont sytstématiquement marqués dans P. Le graphe étant connexe, cette la boucle visite en fait tous les sommets du graphe à l'aide d'un parcours en profondeur. Le tableau P retient le sommet par lequel on est arrivé la première fois à un sommet. La ligne 9 sert à ignorer l'arc par lequel le sommet courant a été découvert.
L'algorithme retourne "false" ssi on peut atteindre le sommet par un autre chemin, c'est-à-dire si le graphe (qui est orienté) contient un cycle.
Le nom de l'algorithme pourrait être IsAcyclic(G).

3. **(2 pts)** Quelle est la complexité d'AlgoSansNom ? Justifiez-votre réponse, éventuellement en an-

notant l'algorithme.

Réponse :

Si le graphe est représenté par une liste d'adjacence, le calcul est le suivant :
- La boucle while au pire exactement $|V|$ itérations. La boucle for, qui itère sur les voisins de chaque sommet, effectuera donc au pire $\sum_{x \in V} \deg(x) = 2|E| = O(|E|)$ itérations.
- On a donc un nombre d'opérations égal à $O(|V| + |E|) = O(|E|)$ puisque le graphe est connexe.

Si le graphe est représenté par matrice d'adjacence, il faut prendre en compte que le calcul de NeighboursOf(v) doit parcourir toute une ligne de la matrice et se fait donc en $\Theta(|V|)$. Comme il est répété au pire $|V|$ fois, il faut ajouter $O(|V|^2)$ à la complexité précédente. On a donc une complexité de $O(|V|^2 + |E|) = O(|V|^2)$.

2 Jeu de billes pour apprendre les carrés (5 points)

Voici un jeu pour deux joueurs, utilisant un sac de n billes.

Les joueurs retirent chacun leur tour un nombre de billes qui doit être un carré ($1, 4, 9, 16, 25$...). Le joueur qui retire la dernière bille du sac a perdu.

Voici un exemple de partie commençant avec $n = 24$ billes :

1. le joueur 1 retire 9 billes, il en reste donc 15
2. le joueur 2 retire 4 billes, il en reste donc 11
3. le joueur 1 retire 1 bille, il en reste donc 10
4. le joueur 2 retire 4 billes, il en reste donc 6
5. le joueur 1 retire 4 billes, il en reste donc 2
6. le joueur 2 retire 1 bille, il en reste donc 1
7. le joueur 1 retire la dernière bille, il a donc perdu.

Voici une autre partie, avec autant de billes, où l'autre joueur gagne.

1. le joueur 1 retire 1 bille, il en reste donc 23
2. le joueur 2 retire 9 billes, il en reste donc 14
3. le joueur 1 retire 9 billes, il en reste donc 5
4. le joueur 2 retire 1 bille, il en reste donc 4
5. le joueur 1 retire 1 bille, il en reste donc 3
6. le joueur 2 retire 1 bille, il en reste donc 2
7. le joueur 1 retire 1 bille, il en reste donc 1
8. le joueur 2 retire la dernière bille, il a donc perdu.

1. **(2 pts)** Justifiez que l'un des deux joueurs possède forcément une stratégie gagnante, quel que soit n. (On ne vous demande pas de dire pour quel joueur la stratégie existe.)

Réponse :

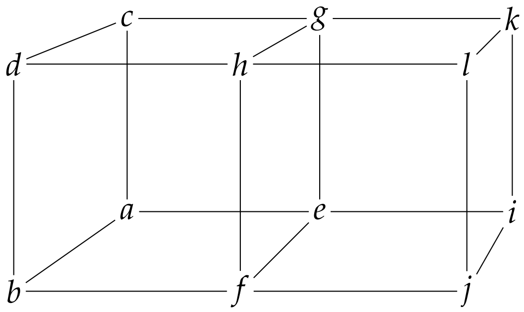
Le jeu peut être représenté par un graphe dans lequel chaque sommet représente le nombre de bille du sac, et tels que deux sommets x et y sont sont reliés ssi $\exists k \in \mathbb{N}$, $x - y = k^2$.

2. **(3 pts)** Quel joueur possède une stratégie gagnante s'il y a $n = 13$ billes initialement ? **Justifiez votre réponse.**

Réponse :

Le noyau du graphe décrit ci-dessus est $N = \{1, 3, 6, 8, 11, 13\}$. Comme $n = 13$ fait partie du noyau, c'est le joueur 2 qui possède la stratégie gagnante (qui consiste à toujours jouer un coup qui amène dans le noyau).

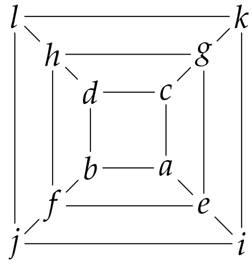
3 Planarité (2 points)



Le graphe ci-dessus est-il planaire ?
Justifiez votre réponse.

Réponse :

Oui. Une justification simple est un dessin :



Une justification encore plus rapide est de dire que ce graphe peut être dessiné sur une sphère sans croiser d'arêtes. On a vu en cours que tout graphe qui peut se dessiner sur une sphère sans croiser d'arêtes est planaire.
En revanche, il est FAUX d'écrire quelque chose comme ceci : le graphe n'a pas de triangle et vérifie $|E| \leq 2|V| - 4$ **donc** il est planaire. Ce *donc* est faux. Ce qu'on a vu en cours c'est tout graphe planaire sans triangle doit vérifier $|E| \leq 2|V| - 4$. Pas l'inverse. On peut construire des graphes non planaires qui vérifient $|E| \leq 2|V| - 4$. (Essayez, ce n'est pas difficile.)
Il est aussi faux d'utiliser la formule d'Euler $|V| - |E| + f = 2$, car la notion de face (f) présuppose que le graphe soit planaire. On ne parle pas des faces d'un graphe avant d'avoir montré qu'il est planaire.

4 Circuits eulériens (10 points)

Nous avons vu en cours une condition pour qu'il existe un cycle eulérien dans un graphe **non-orienté**. Dans cet exercice nous transposons cette notion aux graphes **orientés**.
Voici quelque définitions pour un graphe orienté $G = (V, E)$:

chemin : Un chemin C de G est une succession d'arcs adjacents.
Autrement dit $C = ((s_1, d_1), (s_2, d_2), \dots, (s_n, d_n))$ est un chemin si $(s_i, d_i) \in E$ pour $0 < i \leq n$, et $d_i = s_{i+1}$ pour $0 < i < n$.

circuit : Un circuit de G est un chemin tel que $d_n = s_1$.

circuit eulérien : Un circuit de G est eulérien s'il visite chaque arc du graphe exactement une fois.

graphe équilibré : G est équilibré si le degré entrant de chaque sommet est égal à son degré sortant.

graphe fortement connexe : G est fortement connexe si deux sommets quelconques sont forcément reliés par un chemin (qui respecte le sens des arcs)

graphe connexe : G est connexe si deux sommets quelconques sont reliés dans le graphe non-orienté construit à partir de G en ignorant le sens des arcs.

sommet isolé : Un sommet de G est isolé s'il n'a aucun arc entrant et aucun arc sortant.

1. **(2 pts)** Justifiez que si un graphe sans sommets isolés possède un circuit eulérien, alors ce graphe est équilibré et fortement connexe.

Réponse :

S'il n'y a pas de sommet isolé, tout sommet possède un arc entrant ou sortant et se trouve donc sur le circuit. En suivant ce circuit, on peut alors relier n'importe quels sommets. Le graphe est donc connexe.
Chaque apparition d'un sommet sur le circuit "consomme" un arc entrant et un arc sortant. Si un sommet apparaît n fois dans le circuit, il a donc n arcs entrants et n arcs sortants. Les degrés entrant et sortant de chaque sommet sont donc égaux : le graphe est équilibré.

2. **(3 pts)** Soit G un graphe équilibré, et soit $C = ((s_1, d_1), (s_2, d_2), \ldots, (s_n, d_n))$ un **chemin** de G de longueur la plus grande possible, qui n'utilise pas deux fois le même arc.

   Justifiez que $d_n = s_1$, c'est-à-dire que ce chemin est forcément un circuit.

Réponse :

Notons k le degré sortant de d_n, et notons $(d_n, x_1), \ldots, (d_n, x_k)$ ses arcs sortants.
Ces arcs sortants apparaissent forcément tous dans C, sinon on pourrait allonger C avec un nouvel arc et contredire l'hypothèse du plus long chemin possible.
Considérons un arc sortant (d_n, x_j). Si (d_n, x_j) apparaît dans C à n'importe quelle position (s_i, d_i) pour $i > 1$, il suit un arc (s_{i-1}, d_{i-1}) tel que $d_{i-1} = s_i = d_n$. Si tous ces arcs sortants apparaissent à un position $i > 1$, on a donc que le degré entrant de d_n est $k + 1$ (les prédécesseurs de ces k arcs, plus l'arc (s_n, d_n)), ce qui contredit le fait que le graphe est équilibré.
L'un des arcs sortant apparaît donc en position $i = 1$, et $s_1 = d_n$.

3. **(3 pts)** (Suite de la question précédente.) Soit G un graphe équilibré **sans sommet isolé**, et soit C un chemin de G le plus long possible qui n'utilise pas deux fois le même arc. Nous savons maintenant que C est un circuit.

   Justifiez que si C n'est pas eulérien, alors G n'est pas connexe.

Réponse :

Si C n'est pas eulérien, il existe un arc (s, d) de G qui n'apparaît pas dans le cycle.
Nous avons expliqué dans la réponse précédente que tout arc sortant de d_n apparaissait dans C. Par permutation circulaire, cette propriété est vraie pour les arcs sortants et entrant de tous les sommets intervenant dans le cycle.
L'arc (s, d) ne peut donc pas être connecté (directement ou inderectement) au cycle, et le graphe n'est pas connexe.

4. **(2 pts)** En déduire une condition nécessaire et suffisante pour qu'un graphe sans sommet isolé possède un circuit eulérien.

Réponse :

Un graphe sans sommet isolé possède un circuit eulérien si et seulement si il est connexe est équilibré.

FIN