Université de Nice-Sophia Antipolis
Ecole Polytech' Nice Sophia
SI4
2008–2009

Examen Complexité & algorithmique avancée
du 24 mai 2009

Nom :

Prénom :

Groupe:

Note :

Durée : 2 heures

1	4	
2	8	
3	4	
4	4	
5	3	
6	3	

L'épreuve est composée de cinq questions indépendantes. Veuillez répondre sur la copie avec clarté et concision (le nombre de lignes laissé pour les réponses est parfois excessif). Les réponses *non raisonnables* pourront être pénalisées. L'ensemble des questions représente 25 points et la note sur 20 est la somme des points obtenus (sans dépasser 20).

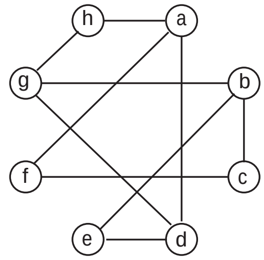
1 Quelques questions rapides

Pour chacune des questions suivantes, indiquez si la réponse est : vraie (V), fausse (F), vraie si et seulement si P= NP(Vsi), vraie si et seulement si P≠ NP(Fsi). Les réponses fausses seront pénalisés, mais le total des points sur l'exercice ne sera pas négative.

	V	F	Vsi	Fsi	
a)					Si $A \propto B$ et $B \in \mathsf{P}$, alors $A \in \mathsf{P}$
b)					Si $A \propto B$ et $B \in \mathsf{NP}$, alors $A \in \mathsf{NP}$.
c)					Tout les problèmes de NP sont NP-complets.
d)					Tout les problèmes NP-complets sont dans NP.
e)					Il existe des problèmes dans NP qui ne sont pas NP-complets.
f)					Un problème NP-complet ne peut pas être résolu en temps polynomial.
g)					Un problème de NP ne peut pas être résolu en temps polynomial.
h)					Si $A \propto B$, $A \in \mathsf{NP}$, et B est NP-complet, alors A est NP-complet.

2 Ensemble dominant

Un *ensemble dominant* D d'un graphe $G(V,E)$ est un ensemble de sommets ($D \subseteq V$), tel que chaque sommet du graphe est soit dans l'ensemble D soit voisin d'un sommet de l'ensemble D.



Dans cet exemple, l'ensemble $\{a,b,c,d,f,g\}$ est un ensemble dominant.
L'ensemble $\{a,b\}$ est un ensemble dominant de cardinalité minimum.

Ici nous proposons une étude détaillé du problème de l'ensemble dominant de cardinalité minimale.

a) Formulez le problème **DS-opt** (*dominating set*) comme un problème d'optimisation.

b) Formulez le problème **DS** (*dominating set*) comme un problème de décision.

c) Prouvez que **DS** $\in$ NP. Expliquez en deux lignes quelles sont les deux options dont vous disposez. Choisissez l'une de ces options et démontrez que **DS** $\in$ NP. On ne vous demande pas de donner tous les détails, mais suffisamment pour évaluer la complexité.

On vous propose la réduction suivante de **X3-SAT** vers **DS** :
Soit $\alpha = C_1 \wedge C_2 \wedge \cdots \wedge C_m$ une formule telle que $C_i = Z_{i,1} \vee Z_{i,2} \vee Z_{i,3}$ où $Z_{i,j}$ est un littéral, qui utilise les variables logiques $X_1, X_2, \cdots, X_n$. Le résultat de la réduction sera composé d'un entier, qui prendra la valeur n et d'un graphe $G(V,E)$ qu'on obtient de la manière suivante : à chaque clause C_i on associe un sommet s_i. Pour chaque variable X_i on crée trois sommets notés x_i, $\neg x_i$ et y_i, reliés entrés eux pour former un triangle. Le sommet qui correspond à une clause est relié au sommet x_i s'il contient X_i comme littéral et à $\neg x_i$ s'il contient $\neg X_i$ comme littéral. Ainsi, chaque sommet qui correspond à une clause est relié à trois sommets qui correspondent à des variables.
d) Appliquez la construction à la formule : $\alpha = (X_1 \vee \neg X_2 \vee X_3) \wedge (\neg X_1 \vee X_2 \vee X_3) \wedge (X_1 \vee X_2 \vee \neg X_3)$.

e) Donnez une formule générale pour le nombre de sommets et le nombre d'arêtes de G en fonction du nombre n de variables de la formule et du nombre m de clauses.

f) Prouvez que la transformation proposée est polynomiale.

g) En utilisant la transformation proposée, prouvez la NP-difficulté de **DS**.

h) Proposez vous-même une réduction de **VC** (*transversale*) vers **DS** et donnez ainsi une seconde preuve de la NP-difficulté de **DS**.

3 Satisfaction multiple

Ce problème concerne une variante de **SAT** où on se demande s'il existe **plusieurs** valeurs de vérité différentes qui satisfont la formule. Nous appellerons ce problème **k-multi-SAT**.

a) Formulez le problème de décision **k-multi-SAT**.

b) Montrez que **k-multi-SAT** $\in$ NP.

c) Prouvez la NP-difficulté de **k-multi-SAT**.
Indication : nous proposons d'utiliser la transformation à partir de **SAT** : si α est une formule de **SAT** et β une formule (constante) en forme normale conjonctive utilisant d'autres variables et qui admet k valeurs de vérité différentes qui la satisfont, alors l'image de la transformation sera $\alpha \wedge \beta$.

4 Algorithme d'approximation

Etant donné un graphe $G(V,E)$ le problème de la coupe maximale consiste en la recherche d'une partition des sommets en S et $T = V - S$ qui maximise le nombre d'arêtes ayant une extrémité dans S et l'autre dans T. Le problème de décision associé est connu comme NP-complet. L'algorithme suivant d'approximation est proposé :

```
Approx-max-cut(G(V,E))
   S ← ∅
   T ← V
   while ∃ v ∈ V tel que si on passe s de S en T ou de T en S
            on augmente le nombre d'arêtes entre S et T
      changer s de coté
   return(S,T)
```

a) Quelle est la complexité (dans le pire des cas) de l'algorithme ? Justifiez.

b) Dans le résultat, pour un sommet quelconque v et pour les arêtes d'extrémité v, comparez le nombre d'arêtes qui sont dans la coupe au nombre d'arêtes qui ne sont pas dans la coupe.

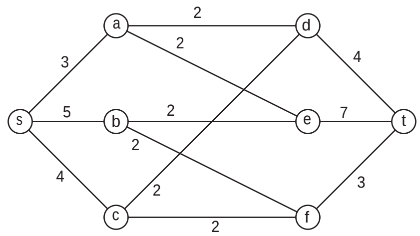
c) Bornez inférieurement le nombre d'arêtes de la coupe obtenue en fonction du nombre d'arêtes du graphe.

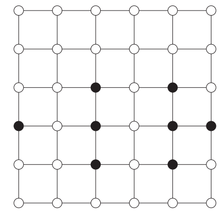
d) Terminez la preuve que l'algorithme est une approximation de ratio ... pour le problème de la coupe maximum.

5 Flots

On souhaite introduire une nouvelle application des flots. Pour cela nous avons besoin d'introduire d'abord un nouveau modèle, dans lequel on ajoute des contraintes de capacité aux noeuds. Cette capacité limite la quantité de flot qui peut traverser le noeud (le flot qui arrive dans le puits ainsi que le flot partant de la source sont aussi considérés comme traversants).
a) Expliquez comment on peut modéliser ce problème pour le ramener au problème du flot classique.

b) Appliquez votre méthode au réseau suivant et trouvez le flot maximum et la coupe de capacité minimum (les noeuds $a - c$ sont de capacité 3).





6 Flots encore

Le problème que nous souhaitons traiter et le suivant : on dispose de k points dans une grille bidimensionnelle $n \times m$: $(x_1,y_1),(x_2,y_2),\cdots,(x_k,y_k)$. On veut chercher des routes sommet et arête disjoints qui permettent d'aller de chaque sommet vers le périmètre de la grille (comme chaque sommet va vers un sommet différent, on peut déduire que $k \leq 2m + 2n - 4$).

c) Proposez une présentation de ce problème en termes de flot avec des capacités sur les arêtes et les sommets.

d) Appliquez votre algorithme à la grille donné plus haut (on doit relier les points noirs au périmètre).