

Année universitaire 2020/21, session 1 d'automne décembre 2020

Mention : master informatique, code UE 4TIN704U

Intitulé de l'épreuve : examen Calculabilité et complexité

Date : 14/12/2020 Heure : 9h00 Durée : 3h

Document autorisé : 1 feuille A4 recto-verso

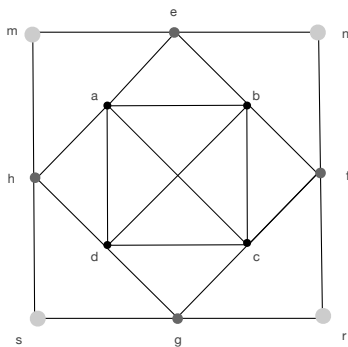
Le barème est indicatif. On attachera une grande importance à la clarté et à la concision des justifications.

Question 1 (16 points)

Dans un graphe non-orienté $G = (V, E)$, une *clique* $C \subseteq V$ est un ensemble de sommets qui sont reliés deux à deux : $uv \in E$ pour tous $u, v \in C, u \neq v$. La distance $d(v, C)$ d'un sommet v par rapport à $C \subseteq V$ est la longueur minimale d'un chemin entre C et v , donc le plus petit ℓ tel que il existe un chemin $v_0, v_1, \dots, v_\ell = v$ avec $v_0 \in C$. On notera cette distance par $d(v, C)$. (Remarque : $v \in C$ si et seulement si $d(v, C) = 0$.)

CLIQUE2 est le problème suivant :

CLIQUE2
Entrée : Un graphe non-orienté $G = (V, E)$ et un entier k .
Sortie : Existe-t-il une clique $C \subseteq V$ de taille k telle que $d(v, C) \leq 2$ pour tout sommet v ?



Dans l'exemple à côté, les sommets a, b, c, d du centre forment une clique C de taille 4, et tous les sommets sont à distance au plus deux de C : les sommets e, f, g, h sont à distance 1, et m, n, r, s à distance 2.

1. (3 p.) Ecrivez un vérificateur polynomial du problème CLIQUE2, en précisant bien quel est le certificat d'une instance positive quelconque et quelle est l'entrée du vérificateur. Quelle est la taille du certificat et quel est le temps de calcul du vérificateur (en fonction de la taille du graphe) ?
2. (1 p.) Que déduisez-vous de la question précédente sur la complexité du problème CLIQUE2 ?
3. (2 p.) Ecrivez un algorithme qui calcule une solution pour le problème CLIQUE2 (s'il en existe). Quelle est sa complexité (en fonction de la taille du graphe) ?

4. (1 p.) Quelle est la différence entre le vérificateur (question 1) et l'algorithme (question 3) pour le problème CLIQUE2?
5. (3 p.) Donnez une réduction polynomiale de CLIQUE2 vers SAT. Vous utiliserez les variables $x_{v,i}$, pour signifier que le sommet v est le i ème sommet de la clique C .
La formule doit être en CNF, et vous devez expliquer les sous-formules dont elle est composée.
6. (2 p.) Justifiez bien qu'il s'agit d'une réduction à la question 5, en montrant comment associer à chaque solution du problème CLIQUE2 une solution pour SAT, et inversement.
7. (1 p.) Combien de clauses comporte votre formule de la question 5? Justifiez que la réduction est polynomiale.
8. (1 p.) Que déduisez-vous de la réduction vers SAT pour la complexité du problème CLIQUE2? Justifiez votre réponse avec les notions que vous avez apprises en cours ou en TD.
9. (2 p.) Dans cette question vous allez montrer une réduction polynomiale du problème CLIQUE (vu en cours) au problème CLIQUE2.

Soit $G = (V, E)$ un graphe non-orienté et k un entier. On veut savoir si cette instance est positive pour CLIQUE, c'est-à-dire si G possède une clique de taille k .

On définit le graphe $G' = (V', E')$ en rajoutant à G un sommet $v_0 \notin V$ qui est relié à tous les sommets de G :

$$V' = V \cup \{v_0\}, \quad E' = E \cup \{v_0v \mid v \in V\}$$

- (a) Montrez que (G, k) est instance positive de CLIQUE si et seulement si $(G', k + 1)$ est instance positive de CLIQUE2.
- (b) Que déduisez-vous de cette réduction de CLIQUE vers CLIQUE2 pour la complexité du problème CLIQUE2? Justifiez votre réponse avec les notions que vous avez apprises en cours ou en TD.

Question 2 (4 points)

Écrivez un programme **LOOP** qui prend en entrée un entier $n \in \mathbb{N}$ et retourne 1 si $n > 0$ et tous les chiffres dans la représentation décimale de n sont les mêmes, et 0 sinon. Par exemple, si $n = 22222$ ou $n = 555$, alors le programme retourne 1. Par contre, si $n = 0$, $n = 42$ ou $n = 1119$, le programme retourne 0.

Indication : Vous pouvez réutiliser directement les fonctions arithmétiques vues en TD (addition, multiplication, division, modulo,...) dans votre programme. Un appel de $\text{ADD}(x, y)$ par exemple, calcule la somme $x + y$ et la met dans la variable res . On suppose qu'il n'y a pas d'effet de bord, en particulier, l'appel ne modifie pas les arguments. Il faut bien-sûr respecter la syntaxe des programmes **LOOP** (en particulier, il n'y a pas d'instruction *break*).