IFT 2125 A13 **Geña Hahn**

Introduction à l'algorithmique

EXAMEN FINAL

le 9 décembre 2013

Durée: 165 minutes

Value: 45% de la note, 50% de la note prédoc

Directives:

- Une feuille remplie recto-verso est la seule documentation permise.
- Les seuls objets permis sur la surface de travail sont la copie de l'examen, la feuille de notes, et les stylos/crayons/gommes.
- Répondez sur le questionnaire, dans l'espace libre qui suit chaque question. Utilisez les verso des pages comme brouillon. **L'espace aloué n'est aucune indication de la longuer de la réponse! Il est souvent beaucoup trop grand.**
- Sauf indication contraire, aucun point ne sera accordé pour une réponse, correcte ou pas, si elle n'est pas accompagnée d'une justification.
- Notez la différence entre *justifier* (argument rapide et court) et *prouver* ou *démontrer* (argument détaillé).
- Vous pouvez vous servir de résultats vus en cours, en TP ou dans des livres à condition de les énoncer précisément. Vous pouvez aussi vous servir des résulats donnés dans l'annexe.
- Pour répondre à une question, vous pouvez également vous servir de résultats énoncés dans d'autres questions dans l'examen, même si vous ne les avez pas démontrés.
- Voir l'Annexe pour des rappels et des formules.

1. ________ /20	5. ________ /15	9. ________ /20
2. ________ /10	6. ________ /20	10. ________ /15
3. ________ /15	7. ________ /15	11. ________ /15
4. ________ /20	8. ________ /15	12. ________ /20

Total: ________ /200 + ______/20 = ___________

Nom: ________________________________ **Code permanent:** __________________

☐ **cocher si cet examen fait partie de votre examen prédoctoral.**

1. (**20 x 1 point**) Pour chacun des énoncés suivants indiquez s'il est vrai ou non en encerclant OUI ou NON. *AUCUNE JUSTIFICATION N'EST NECESSAIRE.*

(1) $O(2^{2^n}) = O(4^n)$	OUI NON
(2) Un algorithme dont la complexité est dans $O(n \lg n)$ est toujours préférable à un autre qui résoud le même problème en $O(3^n)$.	OUI NON
(3) L'algorithme de Kruskal peut être modifié pour qu'il trouve les composantes connexes d'un graphe.	OUI NON
(4) Pour toute fonction $f : \mathbb{N} \longrightarrow \mathbb{R}^{>0}$, $O(f) = \Omega(f)$.	OUI NON
(5) On peut trouver la s-ème plus grande clé dans un fichier de n clés en temps linéaire.	OUI NON
(6) Dans certains cas, l'algorithme de Dijkstra trouve toutes les distances entre les sommets d'un graphe même si les poids des arêtes sont négatifs.	OUI NON
(7) $f \in O(g)$ si, et seulement si, $g \in \Omega(f)$.	OUI NON
(8) $\Theta(\log_3 n) = \Theta(\ln n)$	OUI NON
(9) En moyenne, Quicksort fait $t(n) \in O(n \lg n)$ comparaisons pour trier un fichier de n clés.	OUI NON
(10) Si u est un point d'articulation d'un graphe G alors $G - u$ a exactement deux composantes connexes.	OUI NON
(11) Si $lim_{n\to\infty} \frac{f(n)}{g(n)} = c,\ c \in \mathbb{R}$, alors $f \in \Theta(g)$.	OUI NON
(12) Si $T(n) = 17T(\lfloor \frac{n}{5} \rfloor) + 10^5$ est une fonction éventuellement non-décroissante alors $T(n) \in \Theta(n^{log_5 17})$.	OUI NON
(13) Soit $T(n) = 5T(\frac{n}{3}) + \theta(n)$, $\theta(n) \in \Theta(n^2)$. Alors $T(n) \in \Theta(n^2)$.	OUI NON
(14) Tout ensemble peut être trié.	OUI NON
(15) Il existe une fonction $f : \mathbb{N} \longrightarrow \mathbb{R}^{>0}$ telle que $O(f) = \Omega(f)$.	OUI NON
(16) $P \subseteq NP$ (P et NP sont les classes de complexité décrites à la fin du cours).	OUI NON
(17) $\Theta(2^{\log_3 n}) = \Theta(2^{\ln n})$	OUI NON
(18) La fouille en profondeur est dans $O(m)$ pour tout graphe avec n sommets et m arêtes.	OUI NON
(19) Dans le pire des cas, Quicksort fait $t(n) \in O(n \lg n)$ comparaisons pour trier un fichier de n clés.	OUI NON
(20) Si $T(n) = 17T(\lfloor \frac{n}{5} \rfloor) + 10^5$ est une fonction éventuellement non-décroissante alors $T(n) \in \Theta(n^{lg 17})$.	OUI NON

2. (**5 + 5 points**) Justifiez DEUX de vos réponses à la question 1. **Votre justification ne vaut rien si elle ne comprend pas l'énoncé de ce que vous justifiez!**

   - Question ....

   - Question ....

3. (**15 points**) Prouvez que $\lg n \notin \Theta(n)$.

4. (**20 points**) Utilisez les récurrences pour trouver et prouver la formule explicite pour $\sum_{i=1}^{n} i^2$.

5. (**15 points**) Expliquez comment modifier l'algorithme de Prim pour qu'il trouve les composantes connexes d'un graphe donné non orienté, sans poids sur les arêtes.

6. (**20 points**) Soit $D = (V, A)$ un graphe orienté (sans boucles ou arcs multiples). Soit $\hat{D} = (V, \hat{A})$ la *fermeture transitive* de D, c'est-à-dire, le graphe orienté obtenu à partir de D on ajoutant un arc uv pour chaque paire de sommets (u, v) non-adjacents tels qu'il existe un chemin (orienté) de u vers v. Expliquez comment adapter l'algorithme de Floyd pour trouver $\hat{D}$. Est-ce que la complexité change? Justifiez votre réponse.

7. **(15 points)** Vous avez une batterie de 5 tests à passer dans un cours. Si vous en réussissez la majorité, vous réussissez le cours, sinon... Quelle est la probabilité de votre réussite du cours si vous savez que la probabilité de réussir un des test est (toujours) 0.8? Vous **devez** utiliser une méthode algoritmique vue en cours.

8. (**15 points**) Soit $f, g : \mathbb{N} \longrightarrow \mathbb{R}^{\geq 0}$ et soit $c \in \mathbb{R}^{>0}$ une constante. Faites **UNE** des deux questions suivantes (préciser laquelle vous faites!)

   - Prouvez que si $lim_{n \to \infty} \frac{f(n)}{g(n)} = 0$ alors $f \in O(g)$ mais $g \notin O(f)$.
   - Prouvez rigoureusement à partir de la définition que $\Theta(cf(n)) = \Theta(f(n))$ en précisant les conditions nécessaires s'il y a lieu (les conventions de l'annexe s'appliquent et il n'est pas nécessaire de les répéter).

9. (**20 points**) Donnez un algorithme qui fait *moins que* $\frac{3}{2}n$ comparaisons pour trouver la plus grande et la plus petite clé dans un fichier de n clés. Prouvez votre réponse. Votre algorithme doit être à base de comparaisons.

10. **(15 points)** Trouvez l'ordre exact de la fonction éventuellement non-décroissante définie par

$$T(n) = 4T(\lfloor \frac{n}{3} \rfloor) - 4T(\lfloor \frac{n}{9} \rfloor) + n$$

(et n'oubliez pas de justifier!).

11. **(15 points)** Soit D le graphe orienté suivant, donné par sa matrice d'adjacence.

0	0	1	0	1	0	0
1	0	0	0	0	1	1
0	0	0	1	1	1	1
1	1	1	0	0	0	0
0	1	0	0	0	1	1
1	0	1	0	0	0	0
0	1	0	0	0	1	0

Dans les deux tableaux suivants mettez les résultats des deux premières des itérations (après l'initialisation) de l'algorithme de Floyd (qui trouve toutes les distances entre les paires de sommets) appliqué à D. Combien d'itérations en tout fera l'algorithme?

Première

Deuxième

12. (**20 points**) Prouvez l'énoncé suivant :

> Pour trouver la bille différente parmi n billes qui se ressemblent, et en même temps trouver si elle est plus lourde ou plus légère que les autres (toutes d'un même poids), il faut au moins $\log_3 2n$ pesées sur une balance à bras classique (qui ne peut que comparer les poids de ce qui est sur ses plateaux).

BONIS (10 + 10 points)

1. Prouvez que $VC(G, k)$ est *NP*- complet si et seulement si $CLIQUE(G, k)$ est *NP*- complet (voir l'annexe si vous ne vous souvenez pas de ces deux problèmes).

2. Prouvez que $3 - COL$ se réduit polynomialement à $k - COL$ pour tout $k \geq 3$.

ANNEXE

Rappel: $\mathbb{N}$ est l'ensemble des entiers non négatifs, $\mathbb{R}$ l'ensemble des réels, $\mathbb{R}^{\geq 0}$ l'ensemble des réels non négatifs et $\mathbb{R}^{>0}$ l'ensemble des réels positifs. Aussi, $\lg n = \log_2 n$ et $\ln n = \log_e n$ (où e est la base du logarithme naturel). Finalement, $\log n$ est le logarithme "générique" : la base peut être n'importe quel $b \in \mathbb{R}^{>1}$. Les fonctions considéré dans cette examen sont de $\mathbb{N}$ dans $\mathbb{R}^{\geq 0}$, au moins pour tout n assez grand (i.e. à partir d'un certain n_0).

(1) Soit R la récurrence homogène

$$\sum_{i=0}^{k} a_i t_{n-i} = 0$$

et soit $p(x)$ le polynôme caractéristique de R, avec les racines distinctes $r_1, \ldots, r_\ell$ de multiplicités respectives $m_1, \ldots, m_\ell$. Alors la solution générale de R est

$$\sum_{i=1}^{\ell} \sum_{j=0}^{m_i - 1} c_{ij} n^j r_i^n.$$

(2) Soit R^* la récurrence non homogène

$$\sum_{i=0}^{k} a_i t_{n-i} = \sum_{i=1}^{s} b_i^n q_i(n)$$

avec $b > 0$ et $q_i(n)$ un polynôme en n de degré d_i, $i = 1, \ldots, s$. Soit $p^*(x)$ le polynôme caractéristique de R^*, avec les racines distinctes $r_1, \ldots, r_\ell$ de multiplicités respectives $m_1, \ldots, m_\ell$,

$$p^*(x) = p(x) \prod_{i=1}^{s} (x - b_i)^{d_i + 1}.$$

Alors la solution générale de R est

$$\sum_{i=1}^{\ell} \sum_{j=0}^{m_i - 1} c_{ij} n^j r_i^n.$$

(3) Soit $t(n)$ une fonction éventuellement non-décroissante donnée par la récurrence

$$t(n) = \ell t(\frac{n}{b}) + cn^k$$

pour $n \geq n_0$, avec $b, k, n_0 \in \mathbb{N},\ b \geq 2,\ n_0 \geq 1,\ n > 0,\ c \in \mathbb{R}^{>0}$. Alors

$$t(n) \in \begin{cases} \Theta(n^k) & \text{quand } b^k > \ell \\ \Theta(n^{\log_b \ell}) & \text{quand } b^k < \ell \\ \Theta(n^k \log n) & \text{quand } b^k = \ell \end{cases}$$

et les relations semblables sont vraies pour

$$t(n) = \ell t(\frac{n}{b}) + f(n),$$

$f(n) \in \Theta(n^k)$ et, *mutatis mutandis*, $f(n) \in O(n^k), f(n) \in \Omega(n^k)$.

Remarque. Quand on change la variable ou le codomaine, les conditions originales ne disparaissent pas mais sont traduites dans le nouveau paradigme.

$STABLE(G,k)$ est le problème :

Données : un graphe $G = (V, E)$, un naturel $k > 0$.
Question : Est-ce que G contient un ensemble S, $|S| \geq k$, tel que $uv \notin E$ pour tout $u, v \in S$?

$CLIQUE(G,k)$ est le problème :

Données : un graphe $G = (V, E)$, un naturel $k > 0$.
Question : Est-ce que G contient un ensemble S, $|S| \geq k$, tel que $uv \in E$ pour tout $u, v \in S$?

$k - COL(G)$ est le problème :

Données : un graphe $G = (V, E)$.
Question : Est-ce que V peut être colorié avec k couleurs par une fonction $c : V \longrightarrow \{0, 1, \ldots, k-1\}$ de façon à ce que si $uv \in E$, alors $c(u) \neq c(v)$?

$VC(G,k)$ est le problème :

Données : un graphe $G = (V, E)$, un naturel $k > 0$.
Question : Est-ce que G contient un ensemble S, $|S| \leq k$, tel que si $uv \in E$ alors $\{u, v\} \cap S \neq \emptyset$?

$HAM(G)$ est le problème :

Données : un graphe $G = (V, E)$.
Question : Est-ce que G contient un cycle hamiltonien?

$PHAM(G)$ est le problème :

Données : un graphe $G = (V, E)$.
Question : Est-ce que G contient une chaîne hamiltonienne?

$SUBSETSUM(G,k)$ est le problème :

Donnée : un multi-ensemble $X = \{x_1, x_2, \ldots, x_n\}$, $x_i \in \mathbb{N}$ pour $i = 1, \ldots, n$ (i.e. répétion permise), un naturel $k > 0$.
Question : Y-a-t-il $Y = \{x_{i_1}, \ldots, x_{i_s}\}$ such that $\sum_{j=1}^{s} x_{i_j} = k$?