Algorithmique
Examen Partiel

Jeudi 7 novembre 2016 14h00–16h00

Documentation autorisée : une feuille A4 recto verso manuscrite. Toute autre documentation et appareil sont interdits. **Les téléphones portables doivent être éteints et déposés dans votre sac et vous devez poser vos sacs et vos vestes à l'avant de la salle avant de commencer.**

Indications : Le barème est donné à titre indicatif seulement. Sauf indication contraire, justifiez vos réponses et expliquez vos algorithmes. Les algorithmes doivent être aussi efficaces que possible et présentés de façon lisible et claire. Cela veut dire que les noms de variables doivent être bien choisis pour représenter ce qu'elles sont censées contenir et que les principales étapes doivent être accompagnées d'explications. Une partie de la note portera sur la clarté de présentation de vos réponses (mais pas leur longueur).

Exercice 1 : (6 points)

Un cycle hamiltonien dans un graphe G est un cycle qui visite chaque sommet du graphe une et une seule fois.

1. Donner un algorithme de type *backtracking* qui prend en entrée un graphe G et retourne un cycle hamiltonien de G s'il en existe un, et retourne `None` sinon. Votre algorithme doit être accompagné d'une explication lisible et claire.
2. Donner la complexité de votre algorithme en fonction du nombre de sommets du graphe.

Exercice 2 : (4 points)

L'algorithme de multiplication Russe est le suivant.

```
def mul(u,v):
  if v==1: return u
  return mul(u*2, v//2) + (v%2)*u
```

Par exemple, pour multiplier 114 × 26

```
   114 x 26
=  228 x 13
=  456 x 6 + 228
=  912 x 3 + 228
= 1824 x 1 + 228 + 912
```

La taille d'un entier est la taille de sa représentation en binaire. La division entière est représentée par `//` et le modulo est représenté par `%`. Les opérations sur les entiers sont faites en binaire. La division entière par 2 et le calcul du modulo 2 se font en temps linéaire. L'addition se fait en temps linéaire dans la taille des entiers.

1. Donner la récurrence pour la complexité de cet algorithme, en fonction de la taille des entiers.
2. Résoudre la récurrence pour donner la complexité de l'algorithme. Justifier votre réponse.

Exercice 3 : (3 points)

Résoudre les récurrences suivantes. (Donner uniquement la réponse.)

1. $T(N) = 4T(N/12) + N^3$
2. $T(N) = 4T(N/3) + \log(N)$
3. $T(N) = 4T(N/4) + N^4$

Exercice 4 : (3 points)

Nous avons vu qu'un problème R est dans la classe $\mathcal{NP}$ s'il existe un prédicat binaire `verif` décidable en temps polynomial tel que $R(x)$ est vrai si et seulement il existe un y tel que verif(x, y) est vrai. Décrire le prédicat `verif` pour les problèmes suivants et justifier qu'il est vérifiable en temps polynomial en décrivant un algorithme et donnant sa complexité.

1. 3Col
2. VertexCover

Exercice 5 : (4 points)

Vrai ou Faux ? Donner une brève justification pour les énoncés Vrais.

1. Si le problème R est $\mathcal{NP}$-difficile, et R est dans $\mathcal{NP}$, alors il existe un algorithme polynomial pour R.
2. Si le problème R est $\mathcal{NP}$-difficile, et R est dans $\mathcal{NP}$, alors tout problème dans $\mathcal{NP}$ a un algorithme polynomial.
3. Si $\mathcal{P}$ est inclus dans $\mathcal{NP}$ alors tout problème dans $\mathcal{NP}$ un algorithme polynomial.
4. Si $\mathcal{NP}$ est inclus dans $\mathcal{P}$ alors tout problème dans $\mathcal{NP}$ un algorithme polynomial.
5. Si 3Sat a un algorithme en temps $O(n^5)$ alors tout problème dans $\mathcal{NP}$ a un algorithme en temps $O(n^5)$.
6. Si 3Col $\leqslant$ 2Col alors 3Col a un algorithme polynomial.
7. Si 2Col $\leqslant$ 3Col alors 3Col a un algorithme polynomial.
8. 2Col est dans $\mathcal{NP}$.
9. Clique est dans $\mathcal{NP}$.
10. Clique $\leqslant$ Circuit-SAT.

Annexe

Master Theorem

Théorème. *(Version du cours) Soit T une fonction vérifiant une relation de récurrence de la forme*

$$T(n) = a\,T(\tfrac{n}{b}) + f(n).$$

Alors :

(cas « f petit ») $af(N/b) = cf(N),\ c > 1 \implies T(n) \in \Theta(n^{\log_b a})$

(cas « f moyen ») $af(n/b) = f(N) \implies T(n) \in \Theta(n^{\log_b a} \log n)$

(cas « f grand ») $af(N/b) = cf(N),\ c < 1 \implies T(n) \in \Theta(f(n))$

Théorème. *(Version du TD) Soit T une fonction vérifiant une relation de récurrence de la forme*

$$T(n) = a\,T(\tfrac{n}{b}) + f(n).$$

Alors :

(cas « f petit ») $f(n) \in O(n^{\log_b a - \varepsilon}) \implies T(n) \in \Theta(n^{\log_b a})$

(cas « f moyen ») $f(n) \in \Theta(n^{\log_b a}) \implies T(n) \in \Theta(n^{\log_b a} \log n)$

(cas « f grand ») $\left.\begin{array}{l} f(n) \in \Omega(n^{\log_b a + \varepsilon}) \\ af(\frac{n}{b}) \leqslant cf(n)\ \ (c < 1) \end{array}\right\} \implies T(n) \in \Theta(f(n))$

Problèmes vus en cours et en TD

CIRCUIT-SAT : étant donné un circuit booléen C, existe-t-il une valuation de ses variables d'entrée telle que la sortie du circuit évalue à VRAI ?

3COL : étant donné un graphe G, ses sommets peuvent-ils être coloriés avec 3 couleurs de sorte que toutes les arêtes ait deux couleurs différentes à leurs extrémités ?

2COL : étant donné un graphe G, ses sommets peuvent-ils être coloriés avec 2 couleurs de sorte que toutes les arêtes ait deux couleurs différentes à leurs extrémités ?

INDEPENDENTSET : étant donné un graphe G et un entier k, G possède-t-il un ensemble *indépendant* de sommets de cardinal k (tel qu'il n'existe pas d'arêtes entre eux) ?

CLIQUE : étant donné un graphe G et un entier k, G possède-t-il une *k-clique* (un sous-graphe isomorphe au graphe complet à k sommets) ?

VERTEXCOVER : étant donné un graphe G et un entier k, G possède-t-il un ensemble *couvrant* C de k sommets (tel que toute arête soit incidente à au moins un des sommets de C) ?

DOMINATINGSET : étant donné un graphe G et un entier k, G possède-t-il un ensemble *dominant* D de k sommets (tel que tout sommet de G soit dans D ou voisin d'un sommet de D) ?

SETCOVER : étant donné une collection d'ensembles $S = \{S_1, \ldots, S_n\}$ et un entier k, existe-t-il une sous-collection de S de cardinal k dont l'union des éléments couvre celle des éléments de S ?

HAMILTONIANCYCLE : étant donné un graphe G, existe-t-il un cycle *hamiltonien* dans G (visitant chaque sommet exactement une fois) ?