Examen Final. Complexité algorithmique. 4 janvier 2011.

Durée : 3h. Notes de cours et documents non autorisés.

Le sujet comprend 6 exercices indépendants. Les exercices 1, 2 et 3 concernent des questions de cours. Vous pouvez écrire en Français ou en anglais / You can write in either French or English.

Penser à bien justifier les réponses ; la notation tiendra compte de la rédaction.
Si vous bloquez sur une question d'un exercice, vous pouvez essayer d'admettre le résultat et de passer aux questions suivantes.

Exercice 1.
Donner les définitions des classes de complexité BPP, RP et ZPP.
Indiquer toutes les inclusions connues entre ces classes, ainsi qu'entre ces classes et P, NP et EXP (on ne demande pas ici de démontrer ces inclusions).

Exercice 2.

1. Donner la définition d'une machine de Turing alternante et d'un langage décidé par une machine de Turing alternante.
2. Énoncer le théorème caractérisant la classe PSPACE à l'aide des machines de Turing alternantes, et le démontrer.

Exercice 3.

1. Donner la définition de P/poly (à l'aide de circuits) et de $DTIME(T(n))/a(n)$, où T et a sont deux fonctions de $\mathbb{N}$ vers $\mathbb{N}$.
2. Donner la démonstration du théorème : $\mathsf{P/poly} = \cup_{c,d} DTIME(n^c)/n^d$.

Exercice 4. *Réduction des erreurs pour* RP
Soit $L \subseteq \{0,1\}^*$ tel qu'il existe une machine de Turing probabiliste (MTP) $\mathcal{M}$ de temps polynomial et un $c > 0$ tels que pour tout x de $\{0,1\}^*$:

1. Si $x \in L$, alors $Pr[\mathcal{M}(x) = 1] \geq n^{-c}$, et
2. Si $x \notin L$, alors $Pr[\mathcal{M}(x) = 1] = 0$.

Montrer alors que pour tout $d > 0$ il existe une MTP $\mathcal{M}'$ de temps polynomial telle que pour tout x de $\{0,1\}^*$:

1. Si $x \in L$, alors $Pr[\mathcal{M}'(x) = 1] \geq 1 - 2^{-n^d}$, et
2. Si $x \notin L$, alors $Pr[\mathcal{M}'(x) = 1] = 0$.

Remarque : attention, on demande ici de démontrer ce résultat, pas de le déduire d'un résultat du cours.

Exercice 5.
On dit qu'un graphe orienté G est *fortement connexe* ssi pour tous sommets v_1, v_2 distincts de G il existe un chemin orienté de v_1 à v_2 et un chemin orienté de v_2 à v_1. On considère le langage FC suivant :

$$FC = \{< G > \ / \ G \text{ graphe orienté fortement connexe}\}.$$

1. Montrer que le langage FC appartient à NL.
2. Montrer que ce langage est NL-complet.

Exercice 6.
On note[1] $\mathsf{E} = \bigcup_c \mathsf{DTIME}(2^{cn})$ et $\mathsf{ESPACE} = \bigcup_c \mathsf{SPACE}(2^{cn})$. On rappelle que $A \leq_T^p B$ signifie qu'il existe une machine de Turing déterministe en temps polynomial avec oracle B qui décide A.

1. Montrer que $\mathsf{E} \neq \mathsf{ESPACE} \implies \mathsf{P} \neq \mathsf{PSPACE}$.
   Indication : on pourra utiliser une technique de padding.
2. Raffiner l'argument précédent pour montrer que $\mathsf{E} \neq \mathsf{ESPACE} \implies \mathsf{P} \neq \mathsf{PSPACE} \cap \mathsf{P/poly}$. *Indication : on pourra utiliser le fait que tout langage unaire est dans* $\mathsf{P/poly}$.
3. Le but de cette question est de montrer la réciproque de la question précédente.
   (a) Soit $L \in \mathsf{P/poly}$ décidé par une machine de Turing $\mathcal{M}$ avec conseil polynomial $(\alpha_n)_n$. Montrer que le langage

   $$U = \{1^{\langle n,i,\alpha_{n,i}\rangle} : \alpha_{n,i} \text{ est le } i\text{-ème bit de } \alpha_n\}$$

   vérifie $L \leq_T^p U$.

   (b) *(attention, cette question est plus difficile)*
   Supposons de plus que $L \in \mathsf{PSPACE}$. Montrer que sur l'entrée n, on peut calculer en espace polynomial en n (et non en la longueur de n) un mot β_n vérifiant la même propriété que α_n : la machine $\mathcal{M}$ avec conseil polynomial $(\beta_n)_n$ décide L.

   (c) Montrer qu'il existe alors un langage unaire $U' \in \mathsf{PSPACE}$ tel que $L \leq_T^p U'$.

   (d) Supposons enfin que $L \notin \mathsf{P}$. Montrer qu'il existe un langage unaire dans $\mathsf{PSPACE} \setminus \mathsf{P}$.

   (e) Conclure.
4. Montrer que $\mathsf{BPP} \subseteq \mathsf{PSPACE}$.
5. En utilisant les questions précédentes, montrer que $\mathsf{P} \neq \mathsf{BPP} \implies \mathsf{E} \neq \mathsf{ESPACE}$.

1. Attention à ne pas confondre avec EXP et EXPSPACE.