DEA Algorithmique '01/02 – Introduction à la complexité

Anca Muscholl
web : http://www.liafa.jussieu.fr/~anca
mel : anca@liafa.jussieu.fr

Répondez par Vrai ou Faux aux questions suivantes. Rappel : Les notions «récursif» et «décidable» sont équivalentes, ainsi que «récursivement-énumerable» et «semi-décidable».

		Vrai	Faux
1	Tout langage régulier est semi-décidable.	□	□
2	On ne sait pas décider pour une machine de Turing M et une entrée w de M si la tête de M dépasse l'espace occupé par w durant le calcul.	□	□
3	On ne sait pas décider pour une machine de Turing M s'il existe une entrée w de M telle que la tête de M dépasse l'espace occupé par w durant le calcul.	□	□
4	Les automates à pile boustrophédons (lisant l'entrée dans les deux sens) acceptent que des langages hors-contexte.	□	□
5	La différence de deux langages décidables est un langage décidable.	□	□
6	L'ensemble des fonctions primitives-récursives est dénombrable.	□	□
7	Si un langage A se réduit à un langage B, et B est régulier, alors A est aussi régulier.	□	□
8	Si un langage A se réduit à un langage B (par une réduction polynomiale) et $B \in \mathbf{NP}$, alors $A \in \mathbf{NP}$.	□	□
9	On sait décider si deux machines de Turing sont équivalentes.	□	□
10	La fonction $f : \mathbb{N} \to \mathbb{N}$ définie pour tout $n \in \mathbb{N}$ par $f(n) = 1$ si Dieu existe et 0 sinon, est une fonction primitive-récursive.	□	□
11	On sait décider si un automate fini accepte tous les mots d'un alphabet donné sauf un (qu'on connaît pas).	□	□
12	Le langage $L_n = \{a_1 \cdots a_n a_n \cdots a_1 \mid a_i \in \{a, b\}\}$ est reconnu par un automate non-déterministe de taille polynomiale en n.	□	□
13	Il existe une réduction de SAT au problème de l'arrêt.	□	□
14	Tout langage récursivement-énumérable peut s'ecrire comme projection d'un langage récursif..	□	□
15	Tout langage récursivement-énumérable peut être reconnu par un automate à deux piles (compteurs).	□	□

Bibliographie :

1. C.H. Papadimitriou, Computational complexity. Addison-Wesley, 1994.
   Un des livres de référence les plus récents sur la complexité. Utilisé en cours.
2. U. Schoening, Gems of theoretical computer science. Springer, 1998.
   Traite des thèmes et des questions plus ou moins célébres de la complexité. Livre type "culture générale".
3. M. Sipser, Introduction to the theory of computation. PWS Publishing Company, 1997.
   Introduction à la calculabilité et complexité.
4. J. Hopcroft, J. Ullman, Introduction to automata theory, languages and computation. Addison-Wesley, 1979.
   Introduction classique à la calculabilité et complexité.

Quelques liens utiles :

1. Machines de Turing :
   http://www.ams.org/new-in-math/cover/turing.html
   MT virtuelle (simulation) : http://www.nmia.com/~soki/turing/
   Alan Turing (histoire de la calculabilité) :
   http://www.turing.org.uk/turing/scrapbook/machine.html
2. Poly : page web de Miklos Santha http://www.lri.fr/~santha/, Lecture notes *Complexite et algorithmes probabilistes.*