Introduction à la Calculabilité

Examen du 24 mai 2002

Livres fermés. Durée : 3h30.

Répondez à chaque question sur une feuille séparée sur laquelle figurent votre nom et votre section. Soyez bref et concis, mais précis.

1. Soit un échiquier infini à deux dimensions (i.e. chaque case peut être indexée par un couple de naturels) sur les cases duquel on dispose un nombre fini ou infini de jetons. L'ensemble de toutes les configurations possibles de l'échiquier est-il dénombrable ? Démontrer rigoureusement ce résultat en prouvant tous les théorèmes utilisés.

2. (a) Prouver que si un langage L est décrit par une grammaire régulière, alors L est régulier.

   (b) Soit l'alphabet $\Sigma = \{a,b,c\}$. On définit sur Σ les deux langages suivants : L_1 est le langage décrit par l'expression régulière $a^*(b \cup a^+c)^*$, et L_2 est le langage décrit par la grammaire régulière suivante :

$$\begin{array}{rcl} S & \to & baA \mid \varepsilon \\ A & \to & aA \mid cS. \end{array}$$

   On demande de :

   i. construire des automates finis non déterministes qui acceptent ces deux langages ;

   ii. déterminiser ces automates ;

   iii. construire un automate qui accepte le complément de L_1 et calculer son intersection avec l'automate déterministe qui accepte L_2. Que peut-on en déduire sur la relation entre L_1 et L_2 ?

3. Soit l'alphabet $\Sigma = \{a,b,c\}$ et le langage suivant :

$$L = \{w \in \Sigma^* \mid |w|_a = \min(|w|_b,|w|_c)\}.$$ [1]

   Est-il possible de construire une grammaire hors-contexte qui génère L ? Si oui, donner une telle grammaire. Sinon, démontrer cette impossibilité en prenant soin d'énoncer tous les théorèmes utilisés.

4. Définir les notions de configuration, de dérivation en une et en plusieurs étapes, de langage accepté ainsi que langage décidé dans le cadre des machines de Turing.

---

1. Pour rappel, $|w|_a$ dénote le nombre de a dans le mot w.

5. (a) Le nombre de Gödel d'un mot $w = w_0 \dots w_l$ défini sur un alphabet Σ comportant k symboles, est le naturel $gd(w)$ égal à :

$$\sum_{i=0}^{l} (k+1)^{l-i} gd(w_i)$$

où l'on a préalablement attribué à chaque élément a de Σ, un naturel $gd(a)$ compris entre 1 et k. Soit maintenant une fonction $f(x,y)$ dont le premier argument x est le nombre de Gödel d'un mot w sur l'alphabet Σ, et dont le second argument est un naturel y. $f(x,y)$ est définie comme étant le nombre de Gödel du mot w encodé par x amputé de ses y derniers symboles. Plus formellement, si l'on écrit $w = w_0 \dots w_l$, $f(x,y)$ est le nombre de Gödel du mot :

$$w' = \begin{cases} w_0 \dots w_{l-y} & \text{si } y \leq l \\ \varepsilon & \text{si } y > l \end{cases}$$

Cette fonction est-elle primitive récursive?

(b) Dans la transformation des machines de Turing vers les fonctions $\mu-$récursives, décrire chacune des cinq fonctions de base utilisées dans la démonstration. En déduire la fonction $\mu-$récursive f correspondant à une machine de Turing M.

6. (a) Définir les classes R et RE. Soit un langage L et appelons son complément $\overline{L}$. Quelles sont les trois positions possibles de L et de $\overline{L}$ par rapport à R et RE? Démontrer tous les résultats utilisés.

(b) Soient une machine de Turing M et une grammaire G. Montrer que le problème consistant à tester si le langage accepté par M est égal au langage généré par G, est indécidable.

7. (a) Dans le cadre du théorème de Cook, décrire les variables propositionnelles qui permettent d'encoder l'exécution d'une machine de Turing non déterministe polynomiale. Combien de variables sont-elles requises?

(b) Montrer que le problème de la partition est NP-complet. Ce problème prend pour arguments un ensemble fini X et un entier positif $s(x)$ pour chaque élément $x \in X$. Il consiste à déterminer s'il existe un sous-ensemble $A \subseteq X$ tel que $\sum_{x \in A} s(x) = \left(\sum_{x \in X} s(x)\right)/2$.

Suggestion : Utiliser une transformation polynomiale du problème de la sommation d'un sous-ensemble (SS) vers le problème de la partition. Pour rappel, une instance de SS fournit un ensemble fini Y, un entier positif $s(y)$ pour chaque élément $y \in Y$ et un naturel « objectif » b. Il faut alors déterminer s'il est possible de trouver un sous-ensemble $B \subseteq Y$ pour lequel $\sum_{y \in B} s(y) = b$.

Étant donnée une instance de SS, on ajoutera deux nouveaux éléments inédits u et v, dont le poids sera respectivement $S + b$ et $2S - b$ (où $S = \sum_{y \in Y} s(y)$), les autres éléments conservant le même poids.