Introduction à la Calculabilité

Examen du 14 janvier 2011

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. (a) L'ensemble des programmes C syntaxiquement corrects est-il
      i. dénombrable ?
      ii. récursivement énumérable (dans la classe RE) ?

      Justifiez.

   (b) Qu'en est-il de l'ensemble des machines de Turing qui s'arrêtent lorsque leur mot d'entrée est le mot vide ? Justifiez.

2. Soit L le langage des mots w construits sur l'alphabet $\{a, b, c\}$ et qui respectent au moins l'une des deux conditions suivantes :

   - w commence et se termine par c;
   - $N_a(w) + N_b(w)$ est un multiple de 3, où $N_\sigma(w)$ dénote le nombre d'apparitions du symbole σ dans le mot w.

   Remarque : Les mots c, cc, $cabcc$ sont des exemples de mots qui commencent et se terminent par c.

   (a) Construisez un automate fini non déterministe qui accepte L.

   (b) Construisez un automate fini déterministe qui accepte L.

3. (a) Énoncez et démontrez la deuxième version du théorème du gonflement pour les langages réguliers.

   (b) Soit L un langage régulier. Le langage L' est défini comme étant le langage des mots qui sont la concaténation d'un nombre pair de mots de L. Formellement, on a

   $$L' = \{w \mid (\exists k \in \mathbb{N})(\exists w_1, w_2, \ldots, w_{2k} \in L)\ w = w_1 w_2 \ldots w_{2k}\}.$$

   Le langage L' est-il nécessairement régulier ? Justifiez.

4. (a) Soit L le langage $\{a^n b^* c^m \mid n, m \in \mathbb{N} \text{ et } (n = 2m \text{ ou } m = 2n)\}$.
      i. Décrivez une grammaire hors-contexte qui accepte L.
      ii. Construisez un automate à pile qui accepte L.
   (b) Dans quelles conditions un automate à pile est-il qualifié de *déterministe* ?
   (c) Le complément d'un langage hors-contexte est-il toujours hors-contexte ? Le complément d'un langage hors-contexte déterministe est-il toujours hors-contexte déterministe ? Existe-t-il des langages hors-contexte qui ne sont pas hors-contexte déterministes ?

5. Démontrez que tout langage accepté par une machine de Turing non déterministe est aussi accepté par une machine de Turing déterministe.

6. (a) La fonction SommeDiviseurs(n), où son argument n est un nombre naturel, renvoie la somme des diviseurs stricts de n, c'est-à-dire la somme des diviseurs entiers positifs de n, n non compris. Montrez que cette fonction est primitive récursive.
   Exemple : SommeDiviseurs(6) $= 1 + 2 + 3 = 6$.
   (b) Un nombre naturel est *parfait* s'il est à la fois non nul et égal à la somme de ses diviseurs stricts. Montrez que le prédicat Parfait(n), vrai si et seulement si n est un nombre parfait, est un prédicat primitif récursif.
   (c) Définissez la notion de *prédicat μ-récursif.*
   (d) Le prédicat Parfait(n) est-il μ-récursif ? Justifiez.

7. Soient M_1, M_2 et M_3 trois machines de Turing acceptant respectivement les langages L_1, L_2 et L_3. Démontrez que le problème consistant à déterminer si $(L_1 \cap L_2) \subseteq L_3$ est indécidable.
   Suggestion : utilisez le problème du langage accepté vide.

8. (a) Soit L un langage de la classe NP. Démontrez qu'il existe une machine de Turing déterministe M et un polynôme $p(n)$ tel que M décide L et est de complexité en temps bornée par $2^{p(n)}$.
   (b) Énoncez le théorème de Cook et expliquez son rôle dans la théorie de la NP-complétude.