Université de Strasbourg **UFR de Mathématiques et Informatique**
Master 1 ILC et ISI **octobre 2013**

Complexité et Calculabilité

Contrôle Continu n°1

Durée : 35 minutes
Responsable : Prof. Christian RONSE
Tous documents en papier autorisés mais non partagés
Calculettes inutiles
Téléphones et appareils électroniques éteints et rangés dans un sac fermé

Justifiez soigneusement vos réponses

(1) Machine de Turing.
Soit $\Sigma_0 = \{a, b, c, \ldots\}$ un alphabet ne contenant pas les symboles $\triangleright$ (début de bande) et $\sqcup$ (blanc).

(a) Soit P le sous-ensemble de Σ_0^* formé des mots dans lesquels chaque symbole de Σ_0 apparaît un nombre pair de fois ; par exemple $aababa \in P$ (4 fois a, 2 fois b, pas d'autre lettre), $bacacb \in P$ (2 fois a, 2 fois b, 2 fois c, pas d'autre lettre), mais $bacacbc \notin P$ (3 fois c).

(b) Soit Q le sous-ensemble de Σ_0^* formé des mots dans lesquels a apparaît un (strictement) plus grand nombre de fois que b (en particulier il faut au moins un a) ; par exemple $acbaba \in Q$ (3 fois a, 2 fois b, $3 > 2$), mais $bacacbc \notin Q$ (2 fois a, 2 fois b, $2 = 2$).

Pour chacun des deux langages P et Q, décrire une machine de Turing sur un langage Σ contenant $\Sigma_0 \cup \{\triangleright, \sqcup\}$, formée par assemblage de machines élémentaires, qui décide ce langage.

(2) Grammaire.
Soit $L = 1 \cdot \{0, 1\}^*$ l'ensemble des mots non-vides sur l'alphabet $\{0, 1\}$, dont la première lettre est 1. Soit $f : L \to L \cup \{0\}$ la fonction donnant $f(u) = v$ s'il existe un entier $n > 0$ tel que u et v sont la notation binaire de n et $n - 1$ respectivement. Par exemple, $f(1) = 0$, $f(10) = 1$, $f(11) = 10$, $\ldots$, $f(10000) = 1111$, $\ldots$, $f(10100) = 10011$, etc.

Décrire une grammaire calculant f, c.-à-d. pour $u \in L$ et $v \in \{0, 1\}^*$, on a $SuS \Longrightarrow^* v$ si et seulement si $v = f(u)$. Attention, la première lettre de v ne peut pas être 0, sauf si v n'a qu'une seule lettre, c.-à-d. si $v = 0$.