Théorie des Langages

Magistère STIC

Partiel du 27 avril 2006

durée 2 heures

Les documents sont autorisés.
Les exercices sont indépendants.
Toutes les réponses devront être correctement justifiées.

1 Fonctions séquentielles

Dans cet exercice, on considère l'alphabet $A = \{0,1\}$. Les entiers seront codés en binaire avec le bit de poids faible à gauche. Donc le mot 10011 représente l'entier 25. On considère la fonction $f : A^* \to A^*$ qui code la fonction $g : \mathbb{N} \to \mathbb{N}$ définie par

$$g(n) = \begin{cases} n/2 & \text{si } n \text{ est pair} \\ 3n+1 & \text{sinon.} \end{cases}$$

Formellement, si un mot $v \in A^*$ code l'entier n alors le mot $f(v)$ code l'entier $g(n)$.

a) Calculer les résiduels de la fonction f. On pourra exprimer chaque résiduel comme ci-dessus par la fonction de $\mathbb{N}$ dans $\mathbb{N}$ qu'il code.

b) En déduire l'automate des résiduels de la fonction f.

2 Automates d'arbres

Dans cet exercice, $\mathcal{F}$ est un ensemble fini de symboles de fonctions avec arités et $\mathcal{X}$ est un ensemble fini de variables.

a) Montrer que l'ensemble des instances closes d'un terme linéaire est un langage d'arbres reconnaissables.

On s'intéresse au problème du complément défini comme suit:

Données des termes $t,t_1,\ldots,t_n \in T(\mathcal{F},\mathcal{X})$

Question existe-t-il une instance close de t qui ne soit pas instance d'un des termes $t_1,\ldots,t_n$?

b) Montrer que le problème du complément est décidable si tous les termes $t,t_1,\ldots,t_n$ sont linéaires. Dans ce cas, peut-on construire un automate d'arbres qui reconnaît les solutions du problème du complément ?

c) Montrer que le problème du complément est décidable si les termes $t_1,\ldots,t_n$ sont linéaires et que le terme t est arbitraire. Dans ce cas, peut-on construire un automate d'arbres qui reconnaît les solutions du problème du complément ?

3 Conjugués

Soit Σ un alphabet fini. Deux mots $w,w' \in \Sigma^*$ sont *conjugués* s'il existe deux mots $u,v \in \Sigma^*$ tels que $w = uv$ et $w' = vu$. Dans la suite, on note $C(w)$ l'ensemble des conjugués du mot $w \in \Sigma^*$. De même, pour un langage $L \subseteq \Sigma^*$, on note $C(L) = \bigcup_{w \in L} C(w)$ l'ensemble des conjugués des mots de L.

a) Donner $C(aabaab)$ et $C(\{a^n b^n \mid n \geq 0\})$.

b) Montrer que si L est un langage rationnel alors $C(L)$ l'est aussi.

c) Soient $K,L \subseteq \Sigma^*$ deux langages. Comparer $C(K \cap L)$ et $C(K) \cap C(L)$. Soit $\# \notin \Sigma$. Montrer que $C(\#K \cap \#L) = C(\#K) \cap C(\#L)$.

d) Soient A et B deux alphabets, $L \subseteq A^*$ et $h : A^* \to B^*$ un morphisme. Comparer $h(C(L))$ et $C(h(L))$. Montrer que si h est alphabétique (i.e. $h(a) \in B \cup \{\varepsilon\}$ pour tout $a \in A$) alors $h(C(L)) = C(h(L))$.

Soient $A = \{a_1, \ldots, a_n\}$, $\overline{A} = \{\overline{a}_1, \ldots, \overline{a}_n\}$ et $\Sigma = A \cup \overline{A}$. Pour $w \in A^*$, on note $\overline{w}$ le mot obtenu en remplaçant chaque lettre $a_i \in A$ par $\overline{a}_i$. On rappelle que le langage de Dyck D_n^* est engendré par la grammaire

$$S \longrightarrow \varepsilon + \sum_{i=1}^{n} a_i S \overline{a}_i S$$

Par ailleurs, si $w \in (\Sigma \cup \{\#\})^*$, on note $\rho(w)$ le réduit de w obtenu par effacements successifs de facteurs $a_i \overline{a}_i$ $(1 \leq i \leq n)$. On rappelle que le langage de Dyck D_n^* est aussi l'ensemble des mots $w \in \Sigma^*$ tels que $\rho(w) = \varepsilon$.

e) Montrer que $\rho(C(\#D_n^*)) = \{\widetilde{\overline{\alpha}} \# \alpha \mid \alpha \in A^*\}$, où $\widetilde{w}$ dénote le miroir du mot w.

f) Montrer que le langage $C(\#D_n^*)$ est algébrique.

g) Montrer que si L est un langage algébrique alors $C(L)$ l'est aussi.
Indication : Utiliser le théorème de Chomsky-Schützenberger.