Langages Formels

Licence Informatique – ENS Cachan

Examen du 28 mai 2008

durée 3 heures

Document autorisé : polycopié du cours.
Les exercices sont indépendants.
Toutes les réponses devront être correctement justifiées.

1 Fonctions séquentielles

On appelle machine de Moore un automate déterministe dont les états génèrent des sorties. Formellement, une machine de Moore est un tuple $\mathcal{B} = (Q,A,B,\delta,q_0,F,\lambda)$ où (Q,A,δ,q_0,F) est un automate déterministe et $\lambda : Q \to B^*$ est la fonction (totale) de sortie. La sémantique de $\mathcal{B}$ est une fonction partielle $[\![\mathcal{B}]\!] : A^* \to B^*$ définie sur un mot $u = a_1a_2 \ldots a_{|u|} \in A^*$ si $\delta(q_0,u) \in F$ et dans ce cas $[\![\mathcal{B}]\!](u) = \lambda(q_0)\lambda(q_1) \ldots \lambda(q_{|u|})$ où $q_i = \delta(q_0,a_1 \ldots a_i)$ pour tout $1 \leq i \leq |u|$.

a) Soient $A = \{0,1\}^2$, $B = \{0,1\}$ et $f : A^* \to B^*$ la soustraction. Formellement, une donnée $(a_1,b_1) \ldots (a_p,b_p)$ est interprétée comme un couple (n,m) d'entiers de représentations binaires (avec bit de poids faible en premier) $a_1 \ldots a_p$ et $b_1 \ldots b_p$. La fonction f est définie sur la donnée (n,m) si $n \geq m$ et le résultat est alors $n - m$ toujours représenté en binaire avec bit de poids faible en premier.
Donner une machine de Moore qui réalise cette soustraction, i.e., qui représente la fonction f.

b) On considère le codage défini par $f(x) = a$ et $f(y) = aba$. Peut-on réaliser le codage $f : \{x,y\}^* \to \{a,b\}^*$ avec une machine de Moore ? Peut-on réaliser le décodage f^{-1} avec une machine de Moore ?

c) Montrer que toute fonction séquentielle pure peut être réalisée par une machine de Moore. Qu'en est-il de la réciproque ?

d) Montrer que toute fonction réalisable par une machine de Moore est séquentielle. Qu'en est-il de la réciproque ?

2 Analyse LL

On considère la grammaire G sur l'alphabet $\Sigma = \{\texttt{id},\texttt{num},:=,[,]\}$ définie par

$$\begin{aligned} \textbf{Stmt} &\to \texttt{id} \mid \textbf{Var} := \textbf{Exp} \\ \textbf{Var} &\to \texttt{id} \mid \textbf{Var}[\textbf{Exp}] \\ \textbf{Exp} &\to \texttt{num} \mid \textbf{Var} \end{aligned}$$

a) Pour chaque variable x, calculer $\mathrm{First}_2(x)$ et $\mathrm{Follow}_2(x)$.

b) Montrer que G n'est pas fortement LL(2).

c) Existe-t-il k tel que G soit une grammaire LL(k) ?

3 Automates d'arbres

Soit $A_p = \{1, \ldots, p\}$, Σ un alphabet et $t : A_p^* \to \Sigma$ un arbre. Une feuille de t est un mot $u \in A_p^*$ tel que $u \in \mathrm{dom}(t)$ et $u \cdot 1 \notin \mathrm{dom}(t)$. Un chemin de t est un mot

$$t(\varepsilon) \cdot i_1 \cdot t(i_1) \cdot i_2 \cdot t(i_1 i_2) \cdots i_n \cdot t(i_1 i_2 \ldots i_n)$$

tel que $i_1 i_2 \ldots i_n$ est une feuille de t. On note $\mathcal{C}(t) \subseteq \Sigma^*$ l'ensemble des chemins de t. Si $L \subseteq T_p(\Sigma)$ est un langage d'arbres, on note $\mathcal{C}(L) \subseteq \Sigma^*$ l'ensemble des chemins des arbres de L.

a) Montrer que si $L \subseteq T_p(\Sigma)$ est un langage d'arbres reconnaissable alors $\mathcal{C}(L) \subseteq \Sigma^*$ est un langage reconnaissable de mots.

Si L est un langage d'arbres, on définit la *clôture par chemins* de L comme l'ensemble $\mathcal{CC}(L) \subseteq T_p(\Sigma)$ des arbres t tels que $\mathcal{C}(t) \subseteq \mathcal{C}(L)$.

b) Montrer que si $L \subseteq T_p(\Sigma)$ est un langage d'arbres reconnaissable alors $\mathcal{CC}(L)$ aussi.

c) Montrer qu'un langage d'arbres $L \subseteq T_p(\Sigma)$ est reconnaissable par un automate déterministe descendant si et seulement si $L = \mathcal{CC}(L)$.

d) Peut-on décider si un langage d'arbres reconnaissable $L \subseteq T_p(\Sigma)$ est clos par chemins, i.e., si $L = \mathcal{CC}(L)$?

4 Automates à pile

On s'intéresse ici à des automates dont l'alphabet de pile Γ est un singleton $\{z\}$.

a) Montrer que le langage $L = \{a^n b^m c \mid n \geq m \geq 1\}$ peut être accepté *par pile vide et état final* par un automate dont l'alphabet de pile est un singleton.

b) Montrer que le langage $L = \{a^n b^m c \mid n \geq m \geq 1\}$ ne peut pas être accepté *par pile vide* par un automate dont l'alphabet de pile est un singleton.