

Examen *Théorie des langages*, 14 mai 2007

Documents autorisés : transparents du cours et notes de TD.

Le barème est indicatif (les notes seront ramenées à 20). On attachera une grande importance à la clarté et à la concision des justifications.

Langages réguliers

(4+3+2 points)

Soit $E = a(b + ab)^* + b^*(a + bb)$.

1. Dans cet exercice on veut construire un automate non-déterministe *sans* transitions-ε qui accepte $\mathcal{L}(E)$, en appliquant la construction de Glushkov.

Rappel : la construction de Glushkov s'applique à l'expression renommée

$$E' = a_1(b_1 + a_2b_2)^* + b_3^*(a_3 + b_4b_5).$$

- (a) Déterminez l'ensemble $\text{first}(E')$ des lettres commençant au moins un mot de $\mathcal{L}(E')$, et l'ensemble $\text{last}(E')$ des lettres terminant au moins un mot de $\mathcal{L}(E')$. Justifiez votre réponse.
 - (b) Déterminez l'ensemble $\text{follow}(E')$ des paires de lettres x, y telles que xy est facteur d'au moins un mot de $\mathcal{L}(E')$. Justifiez votre réponse.
 - (c) Construisez l'automate pour $\mathcal{L}(E')$ et ensuite celui pour $\mathcal{L}(E)$.
2. Proposez une expression rationnelle E_1 *sans-étoile* équivalente à E . Justifiez votre réponse.

Rappel : une expression rationnelle sans-étoile sur l'alphabet Σ est une expression construite à partir des lettres de Σ et de l'ensemble vide, en utilisant les opérations union, produit (concaténation) et complémentaire.

3. Proposez une formule ϕ de la logique du premier ordre (FOL) qui décrit le langage $\mathcal{L}(E)$.

Vous pouvez utiliser les macros $\min(x)$ et $\max(x)$ définies en cours.

Automate d'arbres

(4 points)

Dans cet exercice on considère des arbres binaires sur l'alphabet $\Sigma = \Sigma_0 = \Sigma_2 = \{a, b\}$.

Soit $L \subseteq \mathcal{T}(\Sigma)$ l'ensemble des arbres tels que *chaque* chemin de la racine jusqu'à une feuille est étiqueté par un mot de $a^*(ba)^*b^*$. Construisez un automate d'arbre descendant *déterministe* \mathcal{A} qui reconnaît L .

Langages hors-contexte

(2+2+2+2 points)

Soit $L = \{w \in \{a, b, c\}^* \mid |w|_a = |w|_b + |w|_c\}$.

1. Montrez que L n'est pas reconnaissable par application directe du lemme d'itération (ou calcul des quotients).
2. Montrez que L n'est pas reconnaissable par application des propriétés de clôture des langages reconnaissables et du fait que le langage $\{a^n b^n \mid n \geq 0\}$ n'est pas reconnaissable. Justifiez votre réponse.

3. Montrez que chaque mot non-vide w de L peut s'écrire comme $w = \alpha u \beta v$, avec $u, v \in \{a, b, c\}^*$ des mots appartenant à L et α, β des lettres telles que $(\alpha, \beta) \in \{(a, b), (b, a), (a, c), (c, a)\}$. (Par exemple, si $w = baca$, alors $\alpha = b, \beta = a, u = \epsilon$ et $v = ca$. Si $w = ccabaaab$, alors $\alpha = c, \beta = a, u = caba$ et $v = ab$.)
4. Construisez une grammaire hors-contexte G telle que $\mathcal{L}(G) = L$. Justifiez votre réponse.

Expressions arithmétiques

(2+1+3+3 points)

Dans les expressions arithmétiques en notation polonaise on met le symbole de l'opération avant les opérands et on n'utilise pas de parenthèses. Par exemple, $a + b$ s'écrit $+ a b$, et $(a + b) \times (b + c)$ s'écrit $\times + a b + b c$.

1. Décrivez l'ensemble L des expressions en notation polonaise avec les opérations binaires $+, \times$ et les constantes a, b, c par une grammaire hors-contexte G .
2. Quel est l'arbre de dérivation de $\times + a b + b c$?
3. On considère le langage $\mathcal{T}(G)$ des arbres de dérivation de la grammaire G , associés aux mots de $\mathcal{L}(G)$.
 - (a) Précisez l'alphabet Σ des étiquettes des arbres de $\mathcal{T}(G)$, ainsi que le(s) arité(s) de chaque symbole.
 - (b) Construisez un automate d'arbre montant \mathcal{A} qui reconnaît le langage $\mathcal{T}(G)$.
4. Construisez un automate à pile qui reconnaît le langage L .