ensiie Langages et systèmes formels — 2011/2012

Examen 16/11/2011 — 1h45

Tout appareil électronique est interdit et doit être rangé dans un sac fermé. Les notes sur papier sont autorisées. Lisez l'énoncé (1 page). Les exercices sont indépendants et ne sont pas classés par difficulté : ne perdez pas tout votre temps sur un exercice...

Exercice 1 [Automates]

Donner l'automate minimal déterministe acceptant le langage sur $\{0, 1\}$ correspondant à l'expression :

$$(01 + 10)(00 + 11)^\star(01 + 10)^\star$$

On utilisera et détaillera les techniques de composition, déterminisation et minimisation vues en cours.

Exercice 2 [Automates]

Établir la reconnaissabilité éventuelle du langage $\{ww \mid w \in A^\star\}$ pour les cas suivants (donner selon le cas un automate ou une preuve d'impossibilité) :

1. $A = \{0, 1\}$,
2. $A = \{0\}$.

Exercice 3 [Grammaires]

On considère la grammaire G suivante (notation dite *polonaise inverse*) sur le vocabulaire terminal $\{cte, +, \times\}$, le vocabulaire non terminal $\{S\}$ et donc l'axiome S :

$$\begin{array}{lcl} S & \rightarrow & S\,S\,+ \\ S & \rightarrow & S\,S\,\times \\ S & \rightarrow & cte \end{array}$$

1. Donner les mots dans $\mathscr{L}(G)$ de longueur ≤ 5 et des dérivations qui les génèrent.
2. On note $|w|_{cte}$ le nombre d'occurrences du symbole *cte* dans le mot w. Montrer que pour tout $w \in \mathscr{L}(G)$ on a $|w|_+ + |w|_\times + 1 = |w|_{cte}$. On pourra procéder par récurrence sur la longueur de la dérivation.

Exercice 4 [Grammaires]

On suppose donné un symbole terminal noté v muni d'une valeur « nom de variable » (les variables concrètes seront notées x, y ou z dans la suite). On définit un λ-terme comme : toute variable est un λ-terme, si t est un λ-terme alors $\lambda v.t$ est un λ-terme (qu'on appelle une *abstraction*), si t_1 et t_2 sont deux λ-termes alors $t_1\ t_2$ est un λ-terme (qu'on appelle une *application*). Un λ-terme peut enfin être parenthésé.

On peut représenter n'importe quelle fonction calculable dans cette syntaxe très simple mais ce n'est pas le problème ici.

1. Proposer une grammaire non ambiguë pour générer des λ-termes, sachant que l'application est prioritaire sur l'abstraction (c'est-à-dire que le parenthésage implicite de $\lambda x.y\ z$ doit être $(\lambda x.(y\ z))$ et **non pas** $((\lambda x.y)\ z)$) et qu'elle associe à gauche (le parenthésage implicite de $x\ y\ z$ doit être $((x\ y)\ z)$).
2. Donner un arbre de dérivation par votre grammaire de $\lambda x.\lambda y.\lambda z.y\ (x\ y\ z)$. Est-il unique ?