Grenoble INP - Ensimag 1ère année — Janvier 2018

Théorie des langages 1

Durée : 2h.
Documents : tous les documents sont autorisés.

Nb : le barême est donné à titre indicatif ; la rigueur des preuves et la clarté de la rédaction seront prises en compte dans la notation. Cet examen comporte quatre exercices.

Exercice 1 - Automate produit (4 points)

On considère le vocabulaire $V = \{a\}$ et le langage L constitué des mots de V^* dont la longueur est un multiple de 3 **mais pas** un multiple de 2. On pose $L_1 \stackrel{\text{def}}{=} (aaa)^*$ et $L_2 \stackrel{\text{def}}{=} (aa)^*$; on a donc $L = L_1 \cap \overline{L_2}$.

▷ **Question 1** *Construire des automates reconnaissant L_1 et $\overline{L_2}$.*

▷ **Question 2** *En déduire un automate reconnaissant L.*
Indication : *on pourra se servir de la construction de l'automate produit vu en TD.*

▷ **Question 3** *Donner une expression régulière représentant L en posant le système d'équations associé à l'automate de la question précédente et en le résolvant.*

Exercice 2 - Langage racine (8 points)

Etant donné un langage $L \subseteq V^*$, on définit la racine carrée de L comme le langage

$$\mathrm{R}(L) \stackrel{\text{def}}{=} \{w \in V^* \mid ww \in L\}.$$

Le but de cet exercice est de prouver que si L est régulier, alors $\mathrm{R}(L)$ l'est également. Pour un automate A quelconque reconnaissant un langage L, nous proposons donc la construction d'un automate supposé reconnaître le langage $\mathrm{R}(L)$.

Soit $A = (Q, V, \delta, \{q_0\}, F)$ un automate sans ε-transition. On définit l'automate $R(A) = (Q', V, \eta, I, G)$, où :

— $Q' = Q \times Q \times Q$ est l'ensemble des triplets d'éléments de Q.
— L'ensemble des états initiaux de $R(A)$ est[1] $I = \{\langle q_0, q, q\rangle \mid q \in Q\}$.
— $G = \{\langle q, q_f, q\rangle \mid q \in Q \text{ et } q_f \in F\}$.

1. On notera que $R(A)$ contient donc $|Q|$ états initiaux.

— Pour tous triplets $\langle p_1, p_2, p_3\rangle, \langle q_1, q_2, q_3\rangle$ de Q' et tout symbole $a \in V$, on a $(\langle p_1, p_2, p_3\rangle,\ a,\ \langle q_1, q_2, q_3\rangle) \in \eta$ si et seulement si les trois conditions suivantes sont vérifiées :

1. $(p_1, a, q_1) \in \delta$,
2. $(p_2, a, q_2) \in \delta$,
3. $p_3 = q_3$.

On considère l'automate $A_1 = (Q, V, \delta, \{q_0\}, \{q_2\})$ dont la relation de transition δ est définie dans le tableau suivant :

δ	a	b
q_0	q_1	q_0
q_1	q_2	q_1
q_2	q_0	q_2

▷ **Question 1** *Construire l'automate $R(A_1)$.*
Nb : *on veillera à ne représenter que les états atteignables de $R(A_1)$.*

▷ **Question 2** *Construire un automate minimal équivalent à $R(A_1)$ en détaillant les étapes de la construction.*

On considère maintenant un automate $A = (Q, V, \delta, \{q_0\}, F)$ **quelconque**.

▷ **Question 3** *Montrer que si χ est un chemin d'origine $\langle p_1, p_2, p_3\rangle$ et d'extrémité $\langle q_1, q_2, q_3\rangle$ dans $R(A)$, alors $p_3 = q_3$.*

▷ **Question 4** *Prouver que les deux énoncés ci-dessous sont équivalents :*

1. *Il existe un chemin dans R(A) de trace w, d'origine $\langle p_1, p_2, p_3\rangle$ et d'extrémité $\langle q_1, q_2, p_3\rangle$.*
2. *Il existe dans A deux chemins de trace w, l'un d'origine p_1 et d'extrémité q_1 et l'autre d'origine p_2 et d'origine q_2.*

▷ **Question 5** *En déduire que w est reconnu par $R(A)$ si et seulement si ww est reconnu par A.*

Exercice 3 - Grammaire - Modélisation (4 points)

On s'intéresse ici à un sous-ensemble des expressions en Python, défini par la grammaire suivante :

$$\begin{array}{lcl}
\text{exp} & \rightarrow & \text{or_test} \\
\text{or_test} & \rightarrow & \text{and_test} \mid \text{or_test}\ \texttt{or}\ \text{and_test} \\
\text{and_test} & \rightarrow & \text{not_test} \mid \text{and_test}\ \texttt{and}\ \text{not_test} \\
\text{not_test} & \rightarrow & \text{primary} \mid \texttt{not}\ \text{not_test} \\
\text{primary} & \rightarrow & \texttt{True} \mid \texttt{False} \mid \text{Idf} \\
\text{Idf} & \rightarrow & \texttt{a} \mid \ldots \mid \texttt{z}
\end{array}$$

avec exp l'axiome et V_T= {`or`, `and`, `not`, `True`, `False`, `(`, `)`, `a`, ..., `z` }.

▷ **Question 1** *Donner l'arbre de dérivation de l'entrée :*

`not x and True or y`

▷ **Question 2** *Quelles sont les priorités entre les opérateurs* `not`, `or` *et* `and` *?*

▷ **Question 3** *Les programmeurs ne manipulant pas toujours finement l'ordre de priorité entre le* `or` *et le* `and` *certains langages leur donnent la même priorité mais interdisent d'écrire des expressions mélangeant ces opérateurs au même niveau.*

Par exemple l'expression `x and True or y` *ne sera pas correcte mais on pourra écrire* `(x and True) or y` *ou bien* `x and (True or y)` *suivant le calcul qu'on veut effectuer. Il est aussi possible d'écrire plusieurs fois le même opérateur comme dans* `x and True and y`.

Proposer une grammaire non-ambiguë des expressions booléennes pour ce langage. Pour tous les exemples donnés on vérifiera bien que la grammaire est en accord avec les résultats attendus.

Exercice 4 - Preuve de fermeture (4 points)

Rappel : une substitution s est une fonction de V_1 dans $P(V_2^*)$, telle que $s(x)$ est un langage sur V_2 pour tout x dans V_1. $P(V_2^*)$ dénote l'ensemble des sous-ensembles de V_2^*. La substitution s est dite régulière si $s(x)$ est un langage régulier pour tout x dans V_1. De la même manière s est dite hors-contexte si $s(x)$ est un langage hors-contexte pour tout x dans V_1.

On rappelle l'application d'une substitution s sur un langage L.

$$s(L) = \bigcup_{w \in L} s(w)$$

avec $s(\epsilon) = \{\epsilon\}$ et $s(x.u) = s(x).s(u)$ pour tout x dans V_1 et u dans V_1^*.

▷ **Question 1** *Soit L un langage sur V_1^* et s une substitution de V_1 dans $P(V_2^*)$. On s'intéresse à déterminer si $s(L)$ est régulier, hors-contexte ou d'aucun de ces deux types dans les différents cas donnés ci-dessous. On justifiera les réponses en expliquant comment produire $s(L)$ et/ou en donnant des contre-exemples.*

1. *le langage L est hors-contexte et la substitution s est hors-contexte.*
2. *le langage L est hors-contexte et la substitution s est régulière.*
3. *le langage L est régulier et la substitution s est hors-contexte.*