Examen de théorie des langages

Durée 2h. Tous documents autorisés. Vous apporterez un grand soin à la rédaction.

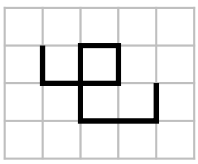
Exercice 1. – Mots avec un facteur interdit.

Soit L le langage composé de tous les mots sur l'alphabet $\{a, b, c\}$ contenant le facteur ac.

a) Donner une expression rationnelle décrivant le langage L.
b) Dessiner un automate fini qui reconnait le langage L.
c) Dessiner un automate fini qui reconnait L^c (le langage complémentaire de L, c'est-à-dire $\{a, b, c\}^* \setminus L$).
d) Dessiner un automate fini déterministe et complet reconnaissant L^c.
e) Par la méthode de votre choix que vous préciserez, calculer et dessiner l'automate minimal du langage L^c.

Exercice 2. – Grammaires et dessins.

Dans cet exercice, on travaille sur l'alphabet $\Sigma = \{\uparrow, \downarrow, \leftarrow, \rightarrow\}$. À un mot $w \in \Sigma^*$ on associe de manière naturelle un dessin $\theta(w)$ dans le plan (à translation près) obtenu en suivant les mouvements décrits successivement par les lettres sans lever le crayon (on peut repasser par un point ou un segment déjà dessiné).

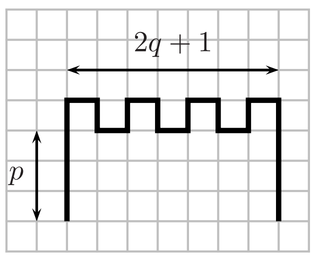


Par exemple, le dessin $\theta(\downarrow\rightarrow\rightarrow\uparrow\leftarrow\downarrow\downarrow\rightarrow\rightarrow\uparrow)$ est dessiné ci-contre.

Étant donné un langage L, on définit $\theta(L) = \{\theta(w) \mid w \in L\}$. On dit qu'une grammaire G engendre un ensemble D de dessins si $\theta(L(G)) = D$.

a) Donner deux mots distincts w_1 et w_2 tels que $\theta(w_1) = \theta(w_2)$.
b) Donner une grammaire qui engendre l'ensemble des segments verticaux de longueur au moins 2.
c) Soit $L_C = \{(\uparrow\rightarrow\downarrow\rightarrow)^n\uparrow\rightarrow\downarrow \mid n > 0\}$. Donner une grammaire qui engendre L_C.

d) Le dessin du château fort $d_{p,q}$ de paramètres p et q est défini par le dessin ci-contre. Donner une grammaire qui engendre $D = \{d_{p,q} \mid p \geqslant 2,\ q \geqslant 1\}$.



Exercice 3. – Langages de l'addition et de la multiplication en unaire.

Sur l'alphabet $\{a, b\}$, on définit le langage de l'addition

$$L_{add} = \{a^m b a^n b a^{m+n} \mid m, n \in \mathbb{N}\}$$

et le langage de la multiplication

$$L_{mul} = \{a^m b a^n b a^{mn} \mid m, n \in \mathbb{N}\}.$$

a) Montrer que le langage L_{add} n'est pas rationnel.
b) Donner une grammaire algébrique qui engendre L_{add}.
c) En utilisant le lemme d'itération (simple), démontrer que le langage L_{mul} n'est pas algébrique.

Exercice 4. – Bord d'un langage.

Étant donné un langage L sur l'alphabet Σ, on définit

$$bord(L) = \{uw \mid \exists v \in \Sigma^*, |u| = |v| = |w| \text{ et } uvw \in L\}.$$

Si L est rationnel, montrer que bord(L) est algébrique. (Indication : expliquer comment construire un automate à pile reconnaissant bord(L). Il n'est pas nécessaire de définir formellement les transitions de l'automate à pile, mais il est impératif d'expliquer la stratégie utilisée.)