

Licence informatique
Parcours recherche et innovation
2015–2016

LF : LANGAGES FORMELS ET CALCULABILITÉ

Examen final

Durée 2h. Notes de cours et de TD autorisées. Les quatre parties sont indépendantes. La plus grande attention sera portée à la qualité de la rédaction, à la rigueur et la précision des argumentations.

Exercice 1 Soient L et L' deux langages sur un alphabet fini Σ .

Question 1.1 On suppose que L est rationnel. A-t-on toujours les implications suivantes ?

1. $L \cup L'$ est rationnel $\Rightarrow L'$ est rationnel
2. $L \cdot L'$ est rationnel $\Rightarrow L'$ est rationnel

Question 1.2 A-t-on toujours l'implication L^* est rationnel $\Rightarrow L$ est rationnel ?

Question 1.3 Reprendre la question 1.1 dans le cas où L est fini.

Exercice 2 Soit $L = \{w \in \{a, b\}^* \mid |w|_a = 0[3] \text{ et } |w|_b = 1[2]\}$, c'est à dire le langage des mots sur $\{a, b\}$ avec un nombre de a égal à 0 modulo 3 et un nombre de b impair.

Question 2.1 Construire un automate fini déterministe reconnaissant L .

Question 2.2 Cet automate est-il minimal ? Si oui, justifier soigneusement votre réponse. Si non, le minimiser.

Exercice 3 Soit $C = \{w\#v \mid w, v \in \{0, 1\}^* \text{ et } w \neq v\}$.

Question 3.1 Construire un automate à pile non-déterministe qui accepte C par état final.

Question 3.2 Montrer qu'il n'existe pas d'automate à pile déterministe qui accepte C par pile vide, c'est à dire, il n'existe pas d'automate à pile \mathcal{A} tel que C soit l'ensemble des mots w tel que l'exécution de \mathcal{A} se termine dans une configuration où la pile est vide.

Question 3.3 Construire une machine de Turing déterministe qui décide C .

Exercice 4

L'objectif est de démontrer le théorème suivant :

Théorème. (de Sheila Greibach)

Soit Γ un alphabet fini. Soit Σ un alphabet fini. Soit $\#$ une lettre qui n'est pas dans Σ . Soit une fonction $L : \Gamma^* \rightarrow P((\Sigma \cup \{\#\})^*)$. Soit $\mathcal{C} = \{L(d) \mid d \in \Gamma^*\}$. On suppose que :

1. $\text{Rat}(\Sigma \cup \{\#\}) \subseteq \mathcal{C}$;
2. (a) Pour tout langage rationnel R , il existe un algorithme f qui, à partir de $d \in \Gamma^*$ calcule $f(d) \in \Gamma^*$ tel que $L(f(d)) = L(d).R$;
 (b) Pour tout langage rationnel R , il existe un algorithme g qui, à partir de $d \in \Gamma^*$ calcule $g(d) \in \Gamma^*$ tel que $L(g(d)) = R.L(d)$;
 (c) Il existe un algorithme h qui, à partir de $d_1, d_2 \in \Gamma^*$, calcule $h(d_1, d_2) \in \Gamma^*$ tel que $L(h(d_1, d_2)) = L(d_1) \cup L(d_2)$.
3. Le problème de décision P_{tout} , défini ci-dessous est indécidable :

P_{tout} entrée : une description $d \in \Gamma^*$ tel que $L(d) \subseteq \Sigma^*$;
 sortie : oui si $L(d) = \Sigma^*$; non sinon.

Soit $\mathcal{P} \subseteq \mathcal{C}$ telle que $\mathcal{P} \neq \emptyset$ et $\mathcal{P} \neq \mathcal{C}$ telle que

- i. $\text{Rat}(\Sigma \cup \{\#\}) \subseteq \mathcal{P}$;
- ii. \mathcal{P} est clos par quotient, c'est à dire : si $L \in \mathcal{P}$ alors $a^{-1}L \in \mathcal{P}$ pour toute lettre $a \in \Sigma \cup \{\#\}$.

Alors le problème $P_{\mathcal{P}}$ défini ci-dessous est indécidable :

$P_{\mathcal{P}}$ entrée : une description $d \in \Gamma^*$;
 sortie : oui si $L(d) \in \mathcal{P}$; non sinon.

Soit $M \subseteq \Sigma^*$ tel que $M \in \mathcal{C}$ et $M \notin \mathcal{P}$. Soit $L \subseteq \Sigma^*$. On note

$$\phi(L) = (\Sigma^* \# M) \cup (L \# \Sigma^*).$$

Question 4.1 Montrer qu'il existe un algorithme A qui, à partir d'une description $d \in \Gamma^*$, calcule $A(d) \in \Gamma^*$ tel que $L(A(d)) = \phi(L(d))$.

Question 4.2 Montrer que $L = \Sigma^*$ si, et seulement si $\phi(L) \in \mathcal{P}$.

Question 4.3 Montrer que $P_{\mathcal{P}}$ est indécidable.

On admet que le problème $P_{\text{Alg}, \text{tout}}$ défini ci-dessous, est indécidable :

$P_{\text{Alg}, \text{tout}}$ entrée : une grammaire algébrique G ;
 sortie : oui si $L(G) = \Sigma^*$; non sinon.

Question 4.4 Choisissez l'un des problèmes suivants (votre favori) et montrer qu'il est indécidable (ils le sont tous) :

$P_{\text{Alg}, \text{Rat}}$ entrée : une grammaire algébrique G ;
 sortie : oui si $L(G)$ est rationnel ; non sinon.

$P_{\text{Alg}, \text{Det}}$ entrée : une grammaire algébrique G ;
 sortie : oui si $L(G)$ est déterministe ; non sinon.

$P_{\text{Alg}, \text{compl}}$ entrée : une grammaire algébrique G ;
 sortie : oui si $\Sigma^* \setminus L(G)$ est algébrique ; non sinon.