Langages formels, calculabilité et complexité
TD8

3 décembre 2015

Exercice 1 Séparation de langages

Soient A et B deux langages disjoints. On dit qu'un langage C *sépare* A et B si $A \subseteq C$ and $B \subseteq \bar{C}$.

1. Si A et $\bar{A}$ ne sont séparés par aucun langage décidable, que peut-on en déduire ?
2. Montrer que A et B sont séparés par un langages décidable si A et B sont co-énumerables et disjoints.
3. Montrer qu'il existe des langages récursivement énumérables qui ne sont séparés par aucun langage décidable.

Exercice 2 Quelques applications

Soient $u_1, \ldots, u_m \in \Sigma^*$. On suppose que $\Sigma \cap \mathbb{N} = \emptyset$. On définit le langage

$$L_u = \{u_{i_1} u_{i_2} \cdots u_{i_n} i_n \cdots i_2 i_1 \mid n \geq 0 \text{ et } 1 \leq i_k \leq m \text{ pour tout } 1 \leq k \leq n\} \ .$$

1. Montrer que L_u et algébrique.

Utiliser l'indécidabilité de PCP pour montrer l'indécidabilité des problèmes suivants :

2. Décider si les langages de deux grammaires hors contexte sont disjoints.

Pour les problèmes suivants, il est utile de considérer le langage

$$L'_u = \{w i_n \cdots i_2 i_1 \mid n \geq 0, w \in \Sigma^*, 1 \leq i_k \leq m \text{ pour tout } 1 \leq k \leq n \text{ et } w \neq u_{i_1} u_{i_2} \cdots u_{i_m}\} \ .$$

3. Montrer que L'_u est algébrique et que $(\Sigma + [m])^* \setminus L_u = L'_u \cup \left((\Sigma + [m])^* \setminus \Sigma^*[m]^*\right)$ où $[m] = \{1, 2, \ldots, m\}$.
4. Décider si deux grammaires hors contexte décrivent le même langage.

Exercice 3 Quines

Pour chaque mot $w \in \Sigma^*$, soit M_w une machine de Turing sur Σ qui écrit le mot w sur le ruban. Pour deux machines de Turing A et B sur Σ, soit $A \cdot B$ une machine de Turing sur Σ qui exécute machine B après avoir exécuté machine A.

1. Expliquer pourquoi la fonction $q : \mathbb{N} \to \mathbb{N}$,

$$q(n) = \begin{cases} \langle M_w \rangle & \text{si } n = \langle w \rangle \text{ avec } w \in \Sigma^* \\ \perp & \text{sinon} \end{cases}$$

   est récursive.

2. Expliquer pourquoi la fonction $s_2 : \mathbb{N} \to \mathbb{N}$,

$$s_2(m, n) = \begin{cases} \langle A \cdot B \rangle & \text{si } m = \langle A \rangle \text{ et } n = \langle B \rangle \\ \perp & \text{sinon} \end{cases}$$

   est récursive.

3. Montrer qu'il existe une machine de Turing M qui écrit $\langle M \rangle$ sur le ruban.

Exercice 4 Fonctions récursives primitives

Montrer que les fonctions suivantes sont récursives primitives.

1. carré(n)
2. $x \mapsto \lfloor \sqrt{x} \rfloor$
3. pndiv(p, n), le nombre de diviseurs de n inférieurs ou égaux à p
4. ndiv(n), le nombre de diviseurs de n
5. Premier(n), retourne 1 si n est premier, 0 sinon
6. la suite de Fibonacci
7. Les fonctions polynomiales
8. La sommation de fonctions récursives primitives.