# Langages Formels

**Exercice 1 — Motifs Objective Caml**
Voici un extrait de la syntaxe des motifs d'Objective Caml, sur l'alphabet terminal $\Sigma = \{\texttt{VN}, \_, ::, \texttt{as}, [, ], ;\}$ et l'ensemble de variables $N = \{\langle pat \rangle, \langle patl \rangle\}$ :

$$\langle pat \rangle \to \texttt{VN} \mid \_ \mid \langle pat \rangle :: \langle pat \rangle \mid \langle pat \rangle\ \texttt{as}\ \texttt{VN} \mid [\ \langle patl \rangle\ ]$$
$$\langle patl \rangle \to \langle pat \rangle \mid \langle patl \rangle\ ;\ \langle pat \rangle$$

Cette grammaire permet de générer des motifs tels que `[a;_;b] :: t as l`

1. Donner une grammaire fortement LL(1) équivalente à ce fragment.
2. Préciser les ensembles $\mathrm{Prem}_1$ et $\mathrm{Suiv}_1$ calculés.
3. Écrire un analyseur récursif descendant dans votre langage favori pour votre grammaire.

**Exercice 2 — Récursivité gauche, ambiguïté**

1. Montrer qu'une grammaire algébrique est LL($k$) si et seulement si, pour toutes dérivations gauches $S \Rightarrow_{lm}^* uA\delta \Rightarrow_{lm} u\alpha_1\delta \Rightarrow_{lm}^* uv_1$ et $S \Rightarrow_{lm}^* uA\delta \Rightarrow_{lm} u\alpha_2\delta \Rightarrow_{lm}^* uv_2$ avec $\mathrm{Prem}_k(v_1) = \mathrm{Prem}_k(v_2)$, nécessairement $\alpha_1 = \alpha_2$.
2. Montrer que si une grammaire est ambiguë, alors elle n'est LL($k$) pour aucun $k$.
3. Une variable $A \in N$ est *récursive gauche* s'il existe une dérivation $A \Rightarrow^+ A\alpha$ avec $\alpha$ dans $(N \uplus \Sigma)^*$. Une grammaire réduite est récursive gauche s'il existe au moins une variable $A$ recursive gauche.

   Montrer que si une grammaire réduite est récursive gauche, alors elle n'est LL($k$) pour aucun $k$.

**Exercice 3 — Langages fortement LL($k$)**

1. Montrer qu'une grammaire algébrique est fortement LL($k$) si et seulement si, pour toutes dérivations gauches $S \Rightarrow_{lm}^* u_1 A \delta_1 \Rightarrow_{lm} u_1 \alpha_1 \delta_1 \Rightarrow_{lm}^* u_1 v_1$ et $S \Rightarrow_{lm}^* u_2 A \delta_2 \Rightarrow_{lm} u_2 \alpha_2 \delta_2 \Rightarrow_{lm}^* u_2 v_2$ avec $\mathrm{Prem}_k(v_1) = \mathrm{Prem}_k(v_2)$, nécessairement $\alpha_1 = \alpha_2$.
2. Montrer qu'une grammaire est LL(1) si et seulement si elle est fortement LL(1).
3. Montrer que la grammaire suivante est LL(2) mais n'est fortement LL($k$) pour aucun $k$ :

$$\begin{aligned} S &\to aAab \mid bAb \\ A &\to cAB \mid \varepsilon \mid a \\ B &\to \varepsilon \end{aligned}$$

4. Montrer que si une grammaire est LL($k$), alors il existe une grammaire fortement LL($k$) équivalente, qui utilise des variables de la forme $(A, F)$ où $F \subseteq \mathrm{Suiv}_k(A)$.
5. Appliquer cette construction à la grammaire précédente.

**Exercice 4 — Calcul efficace de $\mathrm{Prem}_1$ et $\mathrm{Suiv}_1$**

On appelle *expression relationnelle* un terme décrit par la syntaxe abstraite

$$e ::= r \mid e^* \mid e^{-1} \mid e \cdot e \mid e \cup e$$

où $r$ est une relation "atomique". Une expression relationnelle définit une relation $R(e)$. La *taille* $|e|$ d'une expression relationnelle $e$ est la somme des tailles des relations atomiques qui la composent. Pour une expression relationnelle $e$ de taille finie, on peut calculer une image par sa relation $R(e)$ en temps $O(|e|)$. On souhaite calculer les ensembles $\mathrm{Prem}_1$ et $\mathrm{Suiv}_1$ d'une grammaire réduite $G$ à l'aide d'expressions relationnelles de taille $O(|G|)$, de manière à obtenir des algorithmes en temps linéaire $O(|G|)$.

1. Proposer une expression relationnelle $\mathsf{emptyword}$ sur $N \times \{\varepsilon\}$ de taille $O(|G|)$ telle que

$$R(\mathsf{emptyword})(A) = \{\varepsilon \mid A \Rightarrow^* \varepsilon\}\ .$$

2. Proposer une expression relationnelle $\mathsf{prem}$ sur $N \times (\Sigma \uplus \{\varepsilon\})$ de taille $O(|G|)$ telle que

$$R(\mathsf{prem})(A) = \mathrm{Prem}_1(A)\ .$$

3. Proposer une expression relationnelle $\mathsf{end}$ sur $N \times \{\varepsilon\}$ de taille $O(|G|)$ telle que

$$R(\mathsf{end})(A) = \{\varepsilon \mid \exists \delta \in (N \uplus \Sigma)^*,\ S \Rightarrow^* \delta A\}\ .$$

4. Proposer une expression relationnelle $\mathsf{suiv}$ sur $N \times (\Sigma \uplus \{\varepsilon\})$ telle que

$$R(\mathsf{suiv})(A) = \mathrm{Suiv}_1(A)\ .$$

Au moins dans un premier temps, on pourra se contenter d'une expression de taille $O(|G|^2)$.