TD6 : Grammaires algébriques

Exercice 1.

1. Montrer que le langage $\{a^n b^n c^n \mid n \geq 0\}$ n'est pas algébrique, mais qu'il est cependant égal à $L_1 \cap L_2$ où L_1 et L_2 sont algébriques.
2. Montrer que le langage $\{a^m b^m c^n d^n \mid m, n \geq 0\}$ est un langage algébrique qui n'est pas linéaire.

Exercice 2.
Le but de cet exercice est de montrer que $L = \{a^m b^m c^n \mid m, n \geq 0\} \cup \{a^m b^n c^n \mid m, n \geq 0\}$ n'est engendré par aucune grammaire non-ambigüe.

1. Donner une grammaire engendrant ce langage.
2. Déduire du lemme d'Odgen appliqué à $u = a^k b^k c^{k+k!}$ pour un k assez grand, la forme des dérivations possibles de u.
3. En déduire qu'il existe au moins deux dérivations du mot $a^{k+k!} b^{k+k!} c^{k+k!}$.

Exercice 3.
Montrer que la grammaire $S \to$ `if` c `then` S `else` $S|$ `if` c `then` $S|a$ est ambiguë et que le langage engendré n'est pas ambigü.

Exercice 4.
Montrer que tout langage algébrique sans ε-production est engendré par une grammaire algébrique dont toutes les productions sont de la forme $A \to a$, $A \to aB$ et $A \to aBC$.

Exercice 5.
Une grammaire *opérateur* est une grammaire algébrique sans ε-productions et telle qu'il n'existe pas deux symboles consécutifs dans les parties droites des productions qui soient des variables. Montrer que tout langage algébrique sans ε-production a une grammaire opérateur.

Exercice 6.
Une grammaire est en *forme normale de Chomsky* (CNF) si toutes ses règles sont de la forme

$$A \to BC \quad \text{ou} \quad A \to a$$

1. Donner la forme normale de Chomsky de la grammaire $(\{S, A, B\}, \{a, b\}, P, S)$ définie par les productions :

$$\begin{aligned} S &\rightarrow bA|aB \\ A &\rightarrow bAA|aS|a \\ B &\rightarrow aBB|bS|b \end{aligned}$$

2. Montrer que si un mot u est engendré par un non-terminal A, alors il existe des mots u_1 et u_2 respectivement engendrés par B et C tels que $A \rightarrow BC \in P$.
3. En déduire un algorithme en $O(n^3)$ capable de tester si un mot u, où $|u| = n$ est engendré par G.