1 Grammaires non contextuelles

Exercice 1. Donner une grammaire non contextuelle pour chacun des langages suivants (sans trop justifier) :

1. $\mathcal{L}_0 = \{a^i b^j c^k \mid i \neq j \vee j \neq k\}$
2. $\mathcal{L}_1 = \{w \in \{a, b\}^* \mid |w|_a \geq |w|_b\}$
3. $\mathcal{L}_2 = \{w \in \{(,), 0, 1, *, +\} \mid w$ correspond à une expression arithmétique valide$\}$

Exercice 2. Il y a différentes manières de parser les expressions arithmétiques valides. Trouver une grammaire non ambiguë et dans laquelle les arbres de parsing donne la priorité à * par rapport à + (i.e. 2+3*4 est vue comme l'expression qui vaut 14 et non 20).

Exercice 3. Prouver que les grammaires définies par les symboles S et $Balanced$ reconnaissent le même langage mais que l'une est ambiguë et pas l'autre :

$$\begin{array}{rcl} Balanced & \to & (OneRight\ Balanced \mid \epsilon \\ OneRight & \to &) \mid (OneRight\ OneRight \end{array} \qquad\qquad S \to SS \mid (S) \mid \epsilon$$

2 Grammaires sensibles au contexte

Exercice 4. Considérons la grammaire suivante :

$$\begin{array}{rcl} S & \to & aSBC \mid aBC \\ CB & \to & BC \\ aB & \to & ab \end{array} \qquad \begin{array}{rcl} bB & \to & bb \\ bC & \to & bc \\ cC & \to & cc \end{array}$$

Quel langage reconnaît-elle ? Justifier.

Exercice 5. Considérons la grammaire suivante :

$$\begin{array}{rcl} S & \to & CD \\ C & \to & aCA \mid bCB \\ AD & \to & aD \\ BD & \to & bD \\ Aa & \to & aA \end{array} \qquad \begin{array}{rcl} Ab & \to & bA \\ Ba & \to & aB \\ Bb & \to & bB \\ C & \to & \epsilon \\ D & \to & \epsilon \end{array}$$

Quel langage reconnaît-elle ? Justifier.

Exercice 6. Considérons la grammaire suivante :

$$S \to aS \mid aSbS \mid \epsilon$$

Quel langage reconnaît-elle ? Justifier.

3 Myhill – Nerode

Soit Σ un alphabet et $\mathcal{L}_\Sigma$ un langage dessus (pas nécessairement régulier).

Definition 1. *Soient deux mots $x, y \in \Sigma^{*2}$ on définit la relation suivante :*

$$x \sim_{\mathcal{L}_\Sigma} y \quad \stackrel{\text{def}}{=} \quad \forall z \in \Sigma^* : xz \in \mathcal{L}_\Sigma \Leftrightarrow yz \in \mathcal{L}_\Sigma$$

Exercice 7. Montrer que cette relation est bien une relation d'équivalence (i.e. elle est réfléxive, symétrique et transitive).

Exercice 8. Soient $(x, y) \in \Sigma^{*2}$ et $c \in \Sigma$ montrer que si $x \sim_{\mathcal{L}_\Sigma} y$ alors $xc \sim_{\mathcal{L}_\Sigma} yc$.

Exercice 9. Montrer que si $\sim_{\mathcal{L}_\Sigma}$ a un nombre fini de classes d'équivalences, alors $\mathcal{L}_\Sigma$ est régulier.

Exercice 10. Montrer que si $\mathcal{L}_\Sigma$ est régulier alors $\sim_{\mathcal{L}_\Sigma}$ a un nombre fini de classes d'équivalences.

4 Myhill – Nerode : applications

Definition 2. *Calculer les quotients à gauche d'un langage correspond à calculer les classes d'équivalences induites par la relation $\sim_{\mathcal{L}_\Sigma}$ vue plus haut. Les quotients gauches de $\mathcal{L}_\Sigma$ sont les $\mathcal{L}_w$ pour $w \in \Sigma^*$ avec $\mathcal{L}_w = \{x \mid wx \in \mathcal{L}_\Sigma\}$. D'après ce que nous avons vu plus haut les $\mathcal{L}_w$ sont en nombre fini puisque $x \sim_{\mathcal{L}_\Sigma} y \Leftrightarrow \mathcal{L}_x = \mathcal{L}_y$.*

Exercice 11. Trouver les quotients à gauche et l'automate minimal pour chacun des langages suivants sur l'alphabet $\{a, b\}$:

1. $\mathcal{L}_3 = b(ba)^*$
2. $\mathcal{L}_4 = \{a^i b^j \mid i + j \text{ est pair}\}$
3. $\mathcal{L}_5 = \{w \in \{a, b\}^* \mid w \text{ contient exactement une fois le facteur } bb\}$