

A.1 Grammaires et transformations de grammaires

1. Soit la grammaire $S \rightarrow aSb \mid \varepsilon$. Quel est le langage reconnu par cette grammaire? Comment rendre cette grammaire ε -libre?
2. Donner une grammaire qui reconnaisse le langage L (alphabet $X = \{a, b, c\}$).
 $L = \{w \in X^* / w = a^n b^n c^p; n > 0 \text{ et } p > 0\}$
3. Proposer, pour chacun des langages suivants, une grammaire algébrique.
 - langage des expressions arithmétiques (complètement) parenthésées
 $(((4 + -3) * 2), (3 + (4 - (6 * 6))))...$
 - langage des expressions arithmétiques postfixes
 $5\ 4\ +\ 2\ *,\ 3\ 4\ 6\ 6\ * - +...$
 - langage des palindromes : $aabaa, abccba, aa...$
 - $\{w \in X^* / w = a^n b^{2n}; n > 0\}$
 - langage des fonctions d'un langage de programmation, dont les arguments peuvent être des constantes (a, b, c) ou des fonctions (f, g, h) d'arité variable :
 $f(g(a, b), c), a, g(a, g(a), a)...$
 - langage des expressions bien formées de la logique des propositions
 - langage des expressions bien formées de la logique des prédicats
 - $\{w \in X^* / w = a^p b^n c^n a^p; n, p > 0\}$
 - $\{w \in X^* / w = a^n b^p; n \geq p \geq 0\}$
4. Soit la grammaire suivante : $\mathcal{G}_1 = (\{a, b\}, \{S\}, S, \{ S \rightarrow aSbS \mid bSaS \mid \varepsilon \})$.
 - (a) Quel est le langage reconnu?
 - (b) Proposer une grammaire $\mathcal{G}_2$ ε -libre qui reconnaît le même langage.
 - (c) Dessiner les deux arbres de dérivation qui correspondent à l'analyse du mot $aabbaabbbbaab$ au moyen des deux grammaires $\mathcal{G}_1$ et $\mathcal{G}_2$.
5. Soit la grammaire $\mathcal{G}_2 = (\{ (,) \}, \{S\}, S, \{S \rightarrow \varepsilon \mid (S)S\})$. Quel est le langage engendré par cette grammaire? Soit le mot $((()((())))$. Donner la dérivation gauche, la dérivation droite, et l'arbre syntaxique pour ce mot.
6. Transformer la grammaire suivante en la débarrassant des symboles non terminaux inutiles (i.e. sans contribution) :

$$\begin{aligned} S &\rightarrow A \mid B \\ A &\rightarrow aB \mid bS \mid b \\ B &\rightarrow AB \mid Ba \\ C &\rightarrow AS \mid b \end{aligned}$$
7. Soit l'alphabet $X = \{+, =, a\}$. (1) Donner une grammaire algébrique pour le langage L dont chaque mot représente une addition correcte de deux suites de caractères a . Par exemple L contient le mot $aa + aaaa = aaaaaa$. (2) Donner un automate à pile qui reconnaît le même langage.
8. Mettre sous forme normale de Chomsky la grammaire définie par les règles de production suivantes

$$\begin{aligned} S &\rightarrow AB \mid aS \mid a \\ A &\rightarrow Ab \mid \varepsilon \\ B &\rightarrow AS \end{aligned}$$
9. Ambiguïté de grammaires
 - (a) Montrer que la grammaire suivante est ambiguë :
 $S \rightarrow TU; T \rightarrow ST \mid a; U \rightarrow US \mid b$
 - (b) La grammaire suivante est-elle ambiguë? $S \rightarrow aSSb \mid ab$
10. Éliminer la récursivité gauche de la grammaire ETF.
11. Appliquer l'algorithme de dé-récursivation (gauche) vu en cours à la grammaire suivante, grammaire de la liste.

$$\begin{aligned} S &\rightarrow (L) \mid a \\ L &\rightarrow L, S \mid S \end{aligned}$$