

Feuille de travaux pratiques n° 3 Utilisation des grammaires de types 0 à 3

JFLAP est un logiciel permettant d'expérimenter avec les langages formels. Il comporte une section permettant d'utiliser des grammaires rationnelles de type 3, hors-contextes de type 2, ou non-contraintes de type 0. Les grammaires contextuelles de type 1 sont aussi utilisables avec JFLAP mais ne sont pas distinguées des grammaires de type 0.

Exercice 3.1 (Tutoriel sur les grammaires de JFLAP)

JFLAP est téléchargeable depuis le site <http://www.jflap.org/> après inscription. Une copie est disponible sur Madoc dans la section des TP du module X5I0050 – Langages et Automates.

Pour utiliser le programme, il faut le télécharger et le lancer depuis un terminal en tapant `java -jar JFLAP.jar`. Un menu s'affiche alors permettant de sélectionner un des modules du logiciel.

L'utilisation des grammaires correspond à la section *Grammar*. Une fenêtre permettant de saisir une grammaire s'affiche alors. Le terme LHS (Left Hand Side) correspond à la partie gauche des règles et RHS (Right Hand Side) correspond à la partie droite des règles. La partie gauche de la première règle correspond à l'axiome et doit être de la forme $S \rightarrow \dots$.

1. Entrer les règles suivantes : $S \rightarrow bA$, $S \rightarrow a$, $bA \rightarrow b$.
2. Quel est le langage associé à cette grammaire ?
3. Quel est le type de cette grammaire dans la hiérarchie de Chomsky : est-ce une grammaire rationnelle de type 3, une grammaire hors-contexte de type 2, une grammaire contextuelle de type 1 ou non-contraintes de type 0 ?
4. Vérifier que votre réponse est correcte en cliquant sur *Test* puis sur *Test for Grammar Type*.
5. JFLAP peut vérifier qu'une chaîne appartient au langage de cette grammaire et peut afficher l'arbre de dérivation (même dans le cas des grammaires non-contraintes). Pour cela, il faut cliquer sur *Input* puis *Brute Force Parse*. Il faut ensuite entrer une chaîne dans le champ *Input* puis cliquer sur *Start*. Les solutions sont visibles en cliquant plusieurs fois sur *Step*.
 - Essayer avec la chaîne `a` et afficher l'arbre de dérivation.
 - Essayer avec la chaîne `b` et afficher l'arbre de dérivation. Comment est représentée l'utilisation d'une règle **contextuelle** ?

Exercice 3.2 (Un langage complexe)

Le langage $\{a^n b^n c^n, n > 0\}$ n'est ni rationnel, ni algébrique. Il est engendré par la grammaire suivante :

$$\begin{array}{ll} S & \rightarrow AX \\ A & \rightarrow aAbc \\ A & \rightarrow aBbc \\ Bb & \rightarrow bB \\ Bc & \rightarrow D \\ Dc & \rightarrow cD \end{array} \quad \begin{array}{ll} Db & \rightarrow bD \\ DX & \rightarrow EXc \\ BX & \rightarrow \\ cE & \rightarrow Ec \\ bE & \rightarrow Eb \\ aE & \rightarrow aB \end{array}$$

1. Entrer cette grammaire dans JFLAP. Quel est son type ?
2. Vérifier que `aabbcc` fait partie du langage de la grammaire.
3. Dessiner l'arbre de dérivation d'une analyse de ce mot. Combien a-t-il de noeuds ?

Exercice 3.3 (Un langage algébrique)

Le langage $\{a^n b^n, n > 0\}$ est algébrique. Il est engendré par une grammaire hors-contexte.

1. Définir une grammaire engendrant ce langage.
2. Entrer cette grammaire dans JFLAP. Quel est son type ?
3. Vérifier que aaabbb fait partie du langage de la grammaire.
4. Dessiner l'arbre de dérivation d'une analyse de ce mot. Combien a-t-il de noeuds ?
5. Utiliser la recherche `Multiple Brute Force Parse` pour tester plusieurs chaînes en même temps comme aabb, ab, aab, ba.
6. Quels sont les algorithmes d'analyses proposés pour ce type de grammaire.

Exercice 3.4 (Un langage rationnel)

Le langage $\{a^n b^m, n \geq 0, m > 0\}$ est rationnel.

1. Définir une expression rationnelle correspondant à ce langage.
2. Trouver une grammaire de type 3 correspondant à cette expression rationnelle ?
3. Entrer cette grammaire dans JFLAP. Vérifier le type de la grammaire avec JFLAP ?
4. Vérifier dans JFLAP que bbb et aabbb font partie du langage de la grammaire mais que aaa n'est pas accepté.