

Compilation et Implémentation des langages

Corrigé TD0 - Rappels : Grammaires, Langages ...

Enseignante : Hajer SALHI

GLSI – II

Exercice 1 :

Soient les règles de production suivantes :

$$\begin{aligned} A &\rightarrow L := E \\ E &\rightarrow E + T \mid E - T \mid T \\ T &\rightarrow T * P \mid T / P \mid P \\ P &\rightarrow (E) \mid L \\ L &\rightarrow a \mid b \mid c \mid \dots \mid z \end{aligned}$$

1. Ecrire une dérivation droite et une dérivation gauche du mot $a := b * c + d$
2. Dessiner l'arbre syntaxique associé $a := b * c + d$.

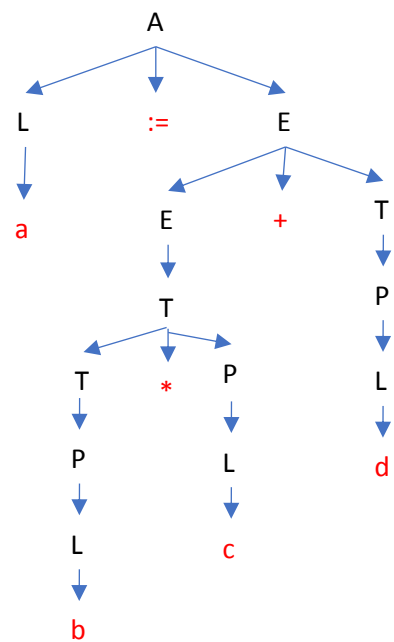
Dérivation Gauche

$A \rightarrow L := E$
 $\rightarrow \underline{a} := E$
 $\rightarrow a := \underline{E + T}$
 $\rightarrow a := \underline{T} + T$
 $\rightarrow a := \underline{T * P} + T$
 $\rightarrow a := \underline{P * P} + T$
 $\rightarrow a := \underline{L * P} + T$
 $\rightarrow a := \underline{b * P} + T$
 $\rightarrow a := b * \underline{L} + T$
 $\rightarrow a := b * \underline{c} + T$
 $\rightarrow a := b * c + \underline{P}$
 $\rightarrow a := b * c + \underline{L}$
 $\rightarrow a := b * c + \underline{d}$

Dérivation droite

$A \rightarrow L := E$
 $\rightarrow \underline{L} := \underline{E + T}$
 $\rightarrow L := E + \underline{P}$
 $\rightarrow L := E + \underline{L}$
 $\rightarrow L := E + \underline{d}$
 $\rightarrow L := \underline{T} + d$
 $\rightarrow L := \underline{T * P} + d$
 $\rightarrow L := T * \underline{L} + d$
 $\rightarrow L := T * \underline{c} + d$
 $\rightarrow L := \underline{P} * c + d$
 $\rightarrow L := \underline{L} * c + d$
 $\rightarrow \underline{L} := b * c + d$
 $\rightarrow \underline{a} := b * c + d$

Arbre Syntaxique



On remarque que pour cet exemple la dérivation gauche et la dérivation droite donnent le même arbre syntaxique

Exercice 2 :

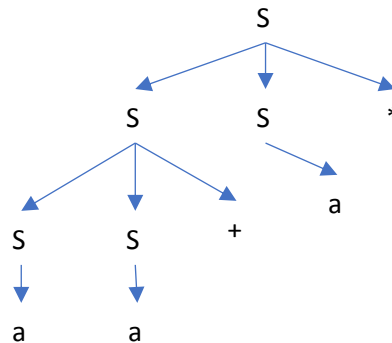
Considérons la grammaire non contextuelle $G = (\{S\}, \{a, +, *\}, S, R)$:

$$R = \{ S \rightarrow S S + \mid S S * \mid a \}$$

1. Montrer comment la chaîne ***aa + a **** est engendrée par cette grammaire.

$$\begin{aligned} S &\rightarrow S S * && \rightarrow \underline{S S} + S * && \rightarrow \underline{a} S + S * \\ &\rightarrow a \underline{a} + S * && \rightarrow a \underline{a} + \underline{a} * \end{aligned}$$

2. Construire un arbre syntaxique correspondant à cette chaîne.



3. Quel langage est engendré par cette grammaire ?

Le langage engendré : les expressions arithmétiques en notation post-fixée utilisant les opérateurs + et *

Exercice 4 :

Quel langage est engendré par les grammaires suivantes ?

- a) $S \rightarrow 0 S 1 \mid 0 1$

Exemples : 01 0011 0000011111 $\rightarrow 0^n 1^n$

Suite de 0 suivie d'une suite de 1 avec le nb de 0 = le nb de 1

Remarque : $0^n 1^n \neq 0^* 1^*$ \rightarrow 0 1 01 000001 01111111

- b) $S \rightarrow + S S \mid - S S \mid a$

Expressions arithmétiques préfixée avec opérateurs + et -

- c) $S \rightarrow a S b S \mid b S a S \mid \epsilon$

Exemples : ab ba abab aabb baba bbaa

nb de a = nb de b

- d) $S \rightarrow a \mid S + S \mid S - S \mid S * S \mid (S)$

Expressions arithmétiques

Exercice 6 :

Décrire les langages dénotés par les expressions régulières suivantes :

- $a(a|b)^*a$ des mots contenant des a et des b commençant par a et se terminant par a
- $((\epsilon|a)^*)^* = a^*$ des suites de a qui peut être vide
- $(a|b)^*a(a|b)(a|b)$ des mots tel que l'avant avant dernier caractère est a
- $a^*b a^*b a^*b a^*$ des mots contenant exactement 3 b
- $(aa|bb)^*((ab|ba)(aa|bb)^*(ab|ba)(aa|bb)^*)^* \rightarrow ((a|b)(a|b))^* \rightarrow$ des mots de longueur paire contenant des a et des b