Université de Nice – Sophia Antipolis | Licence Informatique 3
UE Automates & Langages | 2011–2012

TD n°7

Grammaires non-contextuelles (suite)

Exercice 1) Considérons la grammaire G suivante :

Grammaire G
Axiome = S
N = {S, A, B }
T = {a, b}
P = { S $\to \varepsilon$
S $\to$ a B
S $\to$ b A
A $\to$ a S
A $\to$ b A A
B $\to$ b S
B $\to$ a B B }

1. Quel est le langage $L(G)$ engendré par cette grammaire ?
2. Mettez cette grammaire sous Forme Normale de Chomsky.
3. Cette grammaire est-elle sous Forme Normale de Greibach ?

Exercice 2) On considère le langage L des mots binaires ayant plus de 0 que de 1 (cf. TD n°6 ex.3).

1. Retrouvez une grammaire G pour engendrer ce langage L puis mettez-là sous Forme Normale de Chomsky.
2. A partir de la grammaire obtenue, procédez à l'analyse avec l'algorithme de Cocke, Younger et Kasami du mot : $w = 1001010$.
3. Déduisez-en quels sont les facteurs du mot w qui appartiennent au langage de Dyck.

Exercice 3) On rappelle la définition de l'ensemble $\mathcal{ER}$ des expressions régulières sur l'alphabet Σ (*on omettra volontairement le $\cdot$ symbole de la concaténation*) :

(Base)
- $\varnothing \in \mathcal{ER}$
- "ε" $\in \mathcal{ER}$
- pour tout $\sigma \in \Sigma,\ \sigma \in \mathcal{ER}$

(Induction)
- si $\alpha \in \mathcal{ER}$ et $\beta \in \mathcal{ER}$, alors
  - $(\alpha + \beta) \in \mathcal{ER}$
  - $(\alpha\, \beta) \in \mathcal{ER}$
  - $\alpha^* \in \mathcal{ER}$

1. Précisez l'alphabet du langage $\mathcal{ER}$. Ce langage vous semble-il rationnel ? algébrique non rationnel ?
2. Donnez une grammaire algébrique G qui engendre l'ensemble des expressions régulières sur l'alphabet $\{0, 1\}$.
3. Transformez G en une grammaire G' de telle sorte que G' soit mise sous forme normale de Chomsky.
4. Faites la trace de l'algorithme de CYK sur l'entrée :
$$((1^*0) + 1)$$
Si ce mot est engendré par la grammaire G', vous dessinerez un arbre correspondant à une dérivation pour ce mot.