Théorie des langages et compilation
TD 3

Exercice 1

Soit la grammaire suivante pour définir les expressions bien parenthésées :

$$\begin{array}{lcl} S & \rightarrow & B \ \$ \\ B & \rightarrow & \epsilon \quad | \quad B \ (\ B \ ) \end{array}$$

1. Déterminer les non terminaux qui peuvent se dériver en ϵ.
2. Calculer *premier* et *suivant* pour chaque non terminal.
3. Calculer *directeur* de chaque règle.
4. Cette grammaire est-elle LL(1) ? Justifier.

Soit maintenant la grammaire :

$$\begin{array}{lcl} S & \rightarrow & B \ \$ \\ B & \rightarrow & \epsilon \quad | \quad (\ B \ ) \ B \end{array}$$

5. Déterminer les non terminaux qui peuvent se dériver en ϵ.
6. Calculer *premier* et *suivant* pour chaque non terminal.
7. Calculer *directeur* de chaque règle.
8. Cette grammaire est-elle LL(1) ? Justifier.
9. Dérouler l'algorithme de parsing LL(1) et donner les dérivations correspondantes sur les chaines suivantes :
   (a) (( ) ( )) $
   (b) ( )) ($

Exercice 2

1. Proposer une grammaire LL(1) pour des expressions qui consistent en : des variables, l'addition binaire infixée, la multiplication binaire infixée (avec les règles de priorité usuelles) et les parenthèses.
2. Justifier pourquoi votre grammaire est LL(1).
3. Dérouler l'algorithme de parsing LL(1) et donner les dérivations correspondantes sur les chaines suivantes :
   (a) $id + id * (id + id) \ \$$
   (b) $id * id * id \ \$$
   (c) $((id)) \ \$$

Exercice 3

Soit la grammaire suivante pour définir les chiffres romains (les unités) :

$$
\begin{array}{lcllllllll}
S & \rightarrow & N\,\$ & & & & & & \\
N & \rightarrow & i\,A & | & I_3 & | & v\,I_3 & | & \epsilon \\
A & \rightarrow & v & | & x & & & & \\
I_3 & \rightarrow & I_2\,i & | & \epsilon & & & & \\
I_2 & \rightarrow & I_1\,i & | & \epsilon & & & & \\
I_1 & \rightarrow & i & | & \epsilon & & & &
\end{array}
$$

1. Cette grammaire est-elle ambigüe ? Proposer une grammaire non ambigüe G' équivalente ?
2. G' est-elle LL(1) ?
3. Proposer une grammaire LL(1) équivalente.