IFIPS – Cycle Ingénieur 2e année Informatique
Compilation 2006–2007

Travaux Dirigés n^{o}3
Analyse ascendante

1 Séquence avec séparateur

Soient les grammaires :

$$\begin{array}{|lll|}\hline S & := & L \\ L & := & L\,;A \mid A \\ A & := & a \\ \hline \end{array} \quad \begin{array}{|lll|}\hline S & := & L \\ L & := & A\,;L \mid A \\ A & := & a \\ \hline \end{array} \quad \begin{array}{|lll|}\hline S & := & L \\ L & := & L\,;L \mid A \\ A & := & a \\ \hline \end{array} \quad \begin{array}{|lll|}\hline S & := & L \\ L & := & aT \\ T & := & \epsilon \mid ;L \\ \hline \end{array}$$

1. Montrer que ces grammaires engendrent le même langage.
2. Dire pour chaque grammaire si elle est LR(0) et/ou SLR(1).
3. Comparer la taille de la pile lors de l'analyse ascendante du mot $a; a; a$ par les deux premières grammaires. Quelle remarque peut-on faire ?

2 Différence entre LL(1) et SLR(1)

Soit la grammaire :

$$\begin{array}{|lll|}\hline S & := & AaAb \mid BbBa \\ A & := & \epsilon \\ B & := & \epsilon \\ \hline \end{array}$$

Montrer que cette grammaire est LL(1) mais pas SLR(1).

3 Notation post-fixée

On se donne la grammaire suivante pour reconnaître des expressions arithmétiques écrites de manière post-fixée. L'ensemble des terminaux est {`num`,`op`} et la grammaire possède un seul non-terminal E et les règles suivantes :

$$\begin{array}{|l|}\hline E \coloneqq \texttt{num} \\ E \coloneqq E\ E\ \texttt{op} \\ \hline \end{array}$$

1. Les expressions suivantes appartiennent-elles au langage engendré par cette grammaire ?

(a) `num num num op op`

(b) `num num op num op op`

2. La grammaire est-elle LL(1) ? LR(0) ? SLR(1) ? Dans le cas d'une réponse positive, construire la table de transitions correspondante.
3. On suppose que l'on veut traiter également des opérateurs unaires, pour cela on ajoute la règle E::= E `op`. La nouvelle grammaire ainsi obtenue est-elle LL(1) ? SLR(1) ? LR(1) ?