

# NF11 - Printemps 2013

## Examen final

2 heures

*Documents autorisés*

On rédigera les exercices 1 et 2 d'une part et les exercices 3 et 4 d'autre part.

### Exercice 1 (5 pts)

La grammaire suivante est une grammaire d'expressions arithmétiques préfixées. Les variables sont  $E$ ,  $O$  et  $A$  ; l'axiome  $E$ . Les terminaux sont  $+$ ,  $*$ ,  $nb$ ,  $($ ,  $)$ .

$$\begin{aligned} E &\rightarrow O E E \\ E &\rightarrow A \\ O &\rightarrow + \\ O &\rightarrow * \\ A &\rightarrow nb \\ A &\rightarrow ( E ) \end{aligned}$$

### Questions

1. Montrer que cette grammaire est LL(1).
2. Explicitez les étapes de la reconnaissance de la chaîne :  $*( * 10 5 ) 20$
3. Montrer que l'évaluation de l'expression précédente est 1000.

### Exercice 2 (5 pts)

On considère la grammaire  $G$  suivante.

$$\begin{aligned} S &\rightarrow I S \mid \varepsilon \\ I &\rightarrow b \mid T \\ T &\rightarrow aT \mid \varepsilon \end{aligned}$$

### Question

1. Montrer que la table d'analyse LL(1) de cette grammaire présente trois conflits.
2. Les conflits sur  $S$  et  $I$  viennent de l'erreur de conception :  $S$  et  $I$  nullifiables simultanément.  
Proposer une grammaire  $G'$  très proche de la grammaire  $G$  où  $I$  n'est pas nullifiable.  
Montrer que les conflits sur les variables  $S$  et  $I$  sont retirés.
3. En remarquant que le langage engendré par cette grammaire est le langage  $(a \mid b)^*$  modifier les règles de la grammaire  $G'$  pour obtenir une grammaire LL(1).

**TSVP**

### Exercice 3 (4 pts)

A	B
Nombres : 10, 5, 30	{ "values" : [10, 5, 30], "result" : 45 }

A partir de fichiers ayant la forme A on veut générer des fichiers ayant la forme B (format json) à partir d'une grammaire AntLR et de StringTemplate. La valeur associée à result est la somme des nombres lus ; au moins 1 nombre doit figurer dans la liste.

#### Questions

1. Ecrire une grammaire AntLR permettant de générer un parseur pour le type de fichier A.
2. Ecrire le template de fichier de type B. Donner le type des arguments du template.
3. Ecrire la ou les méthodes du visiteur d'arbre syntaxique qui permet de générer un fichier de type B.

### Exercice 4 (6 pts)

On considère les grammaires G (2 règles) et H (3 règles) suivantes :

$$G : S \rightarrow S b \mid e \quad H : S \rightarrow a S b \mid S b \mid e$$

#### Questions

1. Montrer que la grammaire G est LR(1) ; on notera que le premier état possède 5 items LR(1) à cause de la règle récursive à gauche.
2. Montrer que la grammaire H est ambiguë en étudiant la chaîne ae $\overline{b}b$ .
3. Les états LR(1),  $I_2$  et  $I_8$  de l'automate canonique associé à la grammaire H sont définis par :

$$I_2 = \{ [S \rightarrow a \cdot Sb, \$ / b] [S \rightarrow \cdot aSb, b] [S \rightarrow \cdot Sb, b] [S \rightarrow \cdot e, b] \} \quad I_8 = \delta(I_2, e)$$

Lors de l'analyse de la chaîne ae $\overline{b}b$ , on passe par une situation où sur la pile se trouve  $\overline{8e2a0} \leftarrow$  (fond de pile à droite) et dans le buffer d'entrée  $\overline{bb}$ .

Continuer l'analyse LR(1) de la chaîne et montrer que l'on obtient un conflit réduire-réduire. On sera amené à calculer les items de  $I_8$  et de deux autres états.