

Analyse Syntaxique et Compilation, TD n° 1

Mini-langage d'expressions

Exercice 1 :

Considérez :

$$\begin{aligned}
 b &= \text{let } x = 3 \text{ in} \\
 &\quad \text{let } x = x + x \text{ in} \\
 &\quad \quad \text{let } y = \text{true} \text{ in} \\
 &\quad \quad \quad \text{if } y \text{ then } x + x \text{ else } x
 \end{aligned}$$

Calculez les unités lexicales, l'arbre de dérivation et la syntaxe abstraite de b . Évaluez et typiez b . Compilez b et exécutez le code compilé.

Exercice 2 :

Dans le mini-langage d'expressions arithmétiques et booléennes vu en cours, montrez qu'il y a des expressions qui ne s'évaluent pas en erreur et qui ne sont pas typables.

Exercice 3 :

- Démontrez la proposition suivante vue en cours :

$$\begin{aligned}
 &Si \ x_1 : \tau_1, \dots, x_n : \tau_n \vdash b : \tau, \vdash v_i : \tau_i \text{ pour } i = 1, \dots, n \text{ et} \\
 &[v_1/x_1, \dots, v_n/x_n]b \Downarrow v, \text{ alors } \vdash v : \tau.
 \end{aligned}$$

- Démontrez la version plus forte suivante de cette proposition :

$$\begin{aligned}
 &Si \ x_1 : \tau_1, \dots, x_n : \tau_n \vdash b : \tau \text{ et } \vdash v_i : \tau_i \text{ pour } i = 1, \dots, n, \text{ alors} \\
 &[v_1/x_1, \dots, v_n/x_n]b \Downarrow v \text{ et } \emptyset \vdash v : \tau.
 \end{aligned}$$

Remarque : cette version 'forte' ne se généralise pas à des langages plus compliqués car souvent le typage ne suffit pas à assurer la terminaison du programme évalué.

Analyse LL(1)

Exercice 4 :

Expliquez pourquoi la grammaire suivante n'est pas LL(1) :

$$S \rightarrow iEtSS' \mid a \quad S' \rightarrow eS \mid \epsilon \quad E \rightarrow b$$

La grammaire est-elle ambiguë ?

Exercice 5 :

On considère la grammaire :

$$S \rightarrow A \mid B \quad A \rightarrow aAb \mid 0 \quad B \rightarrow aBbb \mid 1$$

Calculer le FIRST de S , A et B . La grammaire est-elle LL(1)? Pouvez vous construire un automate à pile déterministe qui reconnaît le langage généré par la grammaire ?

Exercice 6 :

On considère la grammaire :

$$\begin{aligned} S &\rightarrow E\$ & E &\rightarrow TE' \\ E' &\rightarrow +TE' \mid \epsilon & T &\rightarrow FT' \\ T' &\rightarrow *FT' \mid \epsilon & F &\rightarrow (E) \mid id \end{aligned}$$

1. Calculez FIRST des symboles non terminaux.
2. Calculez FOLLOW des symboles non terminaux.
3. La grammaire est-elle LL(1)? Expliquez.
4. Calculez une grammaire équivalente sans ϵ -règles.

Analyse LR(0)**Exercice 7 :**

On considère la grammaire G suivante :

$$S \rightarrow C, \quad C \rightarrow 0 \mid aCb .$$

1. Donnez la représentation graphique de l'automate fini non-déterministe qui reconnaît les préfixes admissibles.
2. La grammaire est-elle LR(0)? Expliquez.
3. Dérivez de l'analyse un automate à pile déterministe qui accepte le langage généré par la grammaire.