

Examen

Lundi, 29 mai 2017

Tout document papier est autorisé. Les ordinateurs, les téléphones portables, comme tout autre moyen de communication vers l'extérieur, doivent être éteints et rangés.

Le temps à disposition est de 2 heures. Cet énoncé a 2 pages.

La consultation des copies aura lieu vendredi 2 juin, de 14h à 15h, en salle 128A, bâtiment Condorcet.

Exercice 1 On suppose donnée la définition suivante du type Token :

```
1 public enum Token {A, B, C;}
```

et on considère un fichier jflex qui contient les déclarations suivantes :

```
1 Lettre = [a-z]
2 Voyelle = [aeiou]
```

et les règles lexicales suivantes :

```
1 {Voyelle}+           {return new Token(A);}
2 {Lettre}+{Voyelle}{Lettre}+ {return new Token(B);}
3 {Lettre}*{Voyelle}{Lettre}* {return new Token(C);}
4 [,]                  {}
```

Questions :

- Donner la séquence de jetons produite par le flot d'entrée suivant (il n'y pas d'espaces dans cette entrée, et la fin de la ligne ne fait pas partie de l'entrée) :

```
1 abcde, fghij, oouu, ei
```

- Quel va être le comportement de jflex sur la même entrée quand on change les règles lexicales en

```
1 {Voyelle}+           {return new Token(A);}
2 {Lettre}+{Voyelle}{Lettre}+ {return new Token(B);}
3 {Lettre}*{Voyelle}{Lettre}* {return new Token(C);}
```

- Quel va être le comportement de jflex sur la même entrée quand on change les règles lexicales en

```
1 {Voyelle}+           {return new Token(A);}
2 {Lettre}+{Voyelle}{Lettre}+ {return new Token(B);}
3 {Lettre}*{Voyelle}{Lettre}* {return new Token(C);}
4 [^]                  {}
```

- Pour cette dernière question on vous demande de mettre des règles lexicales dans le bon ordre. Le type des tokens est le même que pour la première question, et les règles lexicales fournies sont

```
1 [zx]+ {return new Token(A);}
2 [xy]+ {return new Token(B);}
3 [yz]+ {return new Token(C);}
4 [,]   {}
```

Mettre les quatre lignes dans un ordre qui assure que jflex lancé sur l'entrée

```
1 yyy, zzz, xzz
```

produise exactement la séquence de jetons

```
1 B C A
```

Exercice 2 On considère la grammaire $G = (V, \Sigma, S, P)$ suivante :

- $V = \{a, b, c, d, \$, S, L, A, D, B\}$
- $\Sigma = \{a, b, c, d, \$\}$,
- $S = S$
- P consiste en les règles suivantes :

$$\begin{aligned} S &\rightarrow L \$ \\ L &\rightarrow A B \mid B \\ A &\rightarrow D \\ D &\rightarrow a \mid c L \\ B &\rightarrow \epsilon \mid d L \mid b \end{aligned}$$

Questions :

1. Dessinez un arbre de dérivation pour la chaîne suivante : `cbdb$`
2. Quels sont les symboles effaçables (ensemble appelé en cours *EPS*) ?
3. Calculez le *Fi* de tous les non-terminaux de la grammaire, en suivant la méthode donnée en cours.
4. Calculez le $FIRST_1$ de tous les côtés droits de la grammaire, en suivant la méthode donnée en cours.
5. Calculez le $FOLLOW_1$ de tous les non-terminaux de la grammaire, en suivant la méthode donnée en cours.
6. Conclure : la grammaire, est-elle LL(1) ?

Exercice 3 On souhaite définir les règles de typage pour des expressions qui mélangent les chaînes de caractères et les entiers, similaire à ce qui existe en Java. Les identificateurs (variables) sont des expressions, ainsi que les constantes entières et les chaînes de caractères constantes. Les expressions peuvent être combinées à l'aide des opérations somme et produit. Formellement, l'ensemble E des expressions est défini comme suit :

- Pour tout $s \in \Sigma^*$, $Ident(s)$ est un élément de E ;
- Pour tout $n \in \mathbb{N}$, $Int(n)$ est un élément de E ;
- Pour tout $s \in \Sigma^*$, $Chaine(s)$ est un élément de E ;
- Si $e_1, e_2 \in E$, alors $Produit(e_1, e_2)$ est un élément de E ;
- Si $e_1, e_2 \in E$, alors $Somme(e_1, e_2)$ est un élément de E .

Un environnement de typage associe des types à des noms des identificateurs, où l'ensemble des types BT est $\{int, chaine\}$.

Un jugement $\Gamma \vdash e : t$, où $e \in E$ est une expression et $t \in BT$ est un type, exprime que e a le type t par rapport à l'environnement Γ .

Le produit ou la somme de deux entiers donne de nouveau un entier. La somme de deux chaînes de caractères donne une chaîne de caractères. De plus, on peut multiplier une chaîne avec un entier, exactement dans cet ordre¹. Compléter les règles d'inférence pour exprimer ces règles de typage (vous les copiez sur votre copie, en remplaçant les points d'interrogations par les bons types).

$$\begin{array}{c} \frac{}{\Gamma \vdash Ident(s) : \Gamma(s)} \\ \frac{}{\Gamma \vdash Chaine(s) : ?} \\ \frac{\Gamma \vdash e_1 : ? \quad \Gamma \vdash e_2 : ?}{\Gamma \vdash Produit(e_1, e_2) : int} \\ \frac{\Gamma \vdash e_1 : ? \quad \Gamma \vdash e_2 : ?}{\Gamma \vdash Produit(e_1, e_2) : chaine} \end{array} \qquad \begin{array}{c} \frac{}{\Gamma \vdash Int(n) : ?} \\ \frac{\Gamma \vdash e_1 : int \quad \Gamma \vdash e_2 : int}{\Gamma \vdash Somme(e_1, e_2) : ?} \\ \frac{\Gamma \vdash e_1 : chaine \quad \Gamma \vdash e_2 : ?}{\Gamma \vdash Somme(e_1, e_2) : ?} \end{array}$$

1. En fait, la somme de deux chaînes est leur concaténation, et le produit d'une chaîne s par un entier n donne n répétitions de s , mais ça n'a aucune importance pour cet exercice car on ne demande que le typage.