

Examen

Jeudi, 3 mai 2018

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.

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

```
1 public enum Token {T1, T2, T3, T4, T5;}
```

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

```
1 Minuscule = [a-z]
2 Majuscule = [A-Z]
3 Lettre = [a-zA-Z]
```

Toutes les questions de cet exercice utilisent le flot d'entrée suivant. Nous précisons qu'il n'y pas d'espaces dans cette entrée, et que la fin de la ligne ne fait pas partie de l'entrée :

```
1 JAVA/javascript/OCaml/pascal/C
```

Questions :

1. Donner la séquence de jetons produite pour les règles lexicales suivantes :

```
1 {Majuscule}+ {return new Token(T1);}
2 {Lettre}+{Majuscule}{Lettre}* {return new Token(T2);}
3 {Minuscule}*{Lettre}{Minuscule}* {return new Token(T3);}
4 [/] {}
```

2. Comment faut-il changer l'ordre des règles lexicales de la question 1 pour que le même flot d'entrée produise la séquence de jetons suivante :

```
1 T2 T3 T2 T3 T3
```

Il suffit de donner les numéros de ligne des règles (par exemple 1234 si vous pensez que l'ordre original donne déjà le résultat demandé).

3. Donner la séquence de jetons produite pour les règles lexicales suivantes :

```
1 {Lettre}+[/]*{Lettre}* {return new Token(T4)}
2 [/] {return new Token(T5)}
```

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

- $\Sigma = \{x, y, z, \$\}$,
- $N = \{S, X, Y, Z\}$
- $S = S$
- P consiste en les règles suivantes :

$$\begin{array}{ll} S \rightarrow X Y z Z \$ & X \rightarrow x X \mid \epsilon \\ Y \rightarrow y Y y \mid \epsilon & Z \rightarrow Y X \end{array}$$

Questions :

1. Dessinez un arbre de dérivation pour la chaîne suivante : xxzyyyy\$

2. Quels sont les symboles effaçables (ensemble appelé en cours *EPS*) ? Il n'est pas demandé de détailler le calcul de l'ensemble *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. La grammaire *G*, est-elle LL(1) ? Justifiez votre réponse.
7. On considère maintenant la grammaire *G'* qui est la même que *G*, sauf que les productions pour *Y* sont remplacées par

$$Y \rightarrow y \ y \ Y \mid \epsilon$$

Expliquez pourquoi *G* et *G'* définissent le même langage. Deux ou trois lignes en Français devraient être suffisantes.

8. Soit acquis que les deux grammaires *G* et *G'* ont les mêmes symboles effaçables, et les mêmes *Fi* pour leurs non-terminaux. Calculez, selon la méthode du cours, pour la grammaire *G'* le *FOLLOW*₁ pour tous les non-terminaux.
9. La grammaire *G'*, est-elle LL(1) ? Justifiez votre réponse.

Exercice 3 On souhaite définir les règles de typage pour des expressions qui mélangent les entiers (type *int*) et les listes d'entiers (type *list*). L'ensemble *E* des expressions de syntaxe abstraite est défini par induction :

- Pour tout $s \in \Sigma^*$, $Var(s)$ est un élément de *E* ;
- Pour tout $n \in \mathbb{N}$, $Int(n)$ est un élément de *E* ;
- *Nil* est un élément de *E* ;
- Si $e_1, e_2 \in E$, alors $Cons(e_1, e_2)$ est un élément de *E* ;
- Si $e_1, e_2 \in E$, alors $Get(e_1, e_2)$ est un élément de *E* ;
- Si $e \in E$, alors $Length(e)$ est un élément de *E*.

Une variable $Var(s)$ peut avoir comme valeur un entier ou une liste. *Nil* est la liste vide, et $Cons(e_1, e_2)$ dénote la liste dont le premier élément est donné par e_1 , et le reste par e_2 . $Length(e)$ retourne la longueur de la liste donnée par e , et $Get(e_1, e_2)$, quand la valeur de e_1 est n_1 , retourne le n_1 -ème élément de la liste donnée par e_2 .

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

1. Est-ce que les expressions suivantes sont bien typées quand x a le type *int* et y a le type *list* ? Justifiez vos réponses.
 - (a) $Get(Var(y), Cons(Int(1), Cons(Int(2), Nil)))$
 - (b) $Get(Length(Cons(Int(42), Nil)), Cons(Get(Var(x), Nil), Cons(Length(Nil), Nil)))$
2. Compléter les règles d'inférence pour exprimer ces règles de typage (vous les copiez sur votre copie, en remplaçant les points d'interrogations).

$\frac{}{\Gamma \vdash Var(s) : ?}$	$\frac{}{\Gamma \vdash Int(n) : ?}$	$\frac{}{\Gamma \vdash Nil : ?}$
$\frac{\Gamma \vdash e_1 : int \quad \Gamma \vdash e_2 : ?}{\Gamma \vdash Cons(e_1, e_2) : ?}$	$\frac{\Gamma \vdash e_1 : ? \quad \Gamma \vdash e_2 : ?}{\Gamma \vdash Get(e_1, e_2) : int}$	$\frac{\Gamma \vdash e : ?}{\Gamma \vdash Length(e) : ?}$