

## Examen partiel du module TRAD1

Aucun document autorisé.

Toute solution illisible ou incompréhensible sera invalide (et sans appel), de même que toute réponse non justifiée.

Toute réponse écrite au crayon de papier sera effacée par nos soins.

### EXERCICE 1 : Ce que vous avez retenu<sup>1</sup>. . . (6 points)

Vous utiliserez la feuille prévue pour cet exercice et vous répondrez directement dessus.

### EXERCICE 2 : Analyse syntaxique. (11 points)

On considère la grammaire augmentée  $G$  suivante avec, comme terminaux les symboles  $a$   $b$   $x$  et comme ensemble de non-terminaux  $\{S', S, A, B\}$ ,  $S'$  étant l'axiome. Dans cette grammaire,  $\wedge$  désigne le mot vide.

$S'$	$\rightarrow$	$S$	$r_0$
$S$	$\rightarrow$	$A$	$r_1$
		$x B a$	$r_2$
$A$	$\rightarrow$	$a A B$	$r_3$
		$\wedge$	$r_4$
$B$	$\rightarrow$	$b B$	$r_5$
		$x$	$r_6$

Est-ce qu'un analyseur  $LALR(1)$  de cette grammaire peut reconnaître l'entrée suivante :  $aabxx$  ?

Construisez toutes les structures nécessaires à la justification de votre réponse.

### EXERCICE 3 : Fonctions sémantiques. (3 points)

On considère la grammaire  $G$  suivante avec, comme terminaux les symboles  $,$   $($   $)$  *nbreel* et comme ensemble de non-terminaux  $\{A, S, V\}$ ,  $A$  étant l'axiome. Dans cette grammaire,  $\wedge$  désigne le mot vide.

$A$	$\rightarrow$	$( V , V ) S$
$S$	$\rightarrow$	$, A$
		$\wedge$
$V$	$\rightarrow$	<i>nbreel</i>

- Dessinez l'arbre syntaxique de l'entrée :  $(\frac{pi}{2}, 2), (\pi, 1)$
- Ajoutez des fonctions sémantiques permettant d'afficher les coordonnées cartésiennes correspondantes dans le cas d'une analyse syntaxique descendante. On rappelle les formules de conversion :

$$x = r \times \cos \theta$$

$$y = r \times \sin \theta$$

Par exemple, pour  $(\frac{pi}{2}, 2), (\pi, 1)$ , la sortie console doit afficher  $(0, 2), (-1, 0)$ . La notation précédente respecte la convention  $(\theta, r)$ .

<sup>1</sup>Toutes les réponses aux questions de cet exercice ont été données lors des CM, ou se trouvent dans l'article distribué.

Nom, prénom :

Question	VRAI	FAUX
Pour un compilateur multi passes, la sortie de la passe k est utilisée comme entrée de la passe k+1.		
Une grammaire ambiguë ne peut être ni LL(1), ni SLR(1).		
Toute grammaire LR(0) est SLR(1).		
Il existe 3 types de conflits en analyse ascendante : <i>lecture – reduction</i> , <i>reduction – reduction</i> , et <i>lecture – lecture</i> .		
Un interprète est un logiciel exécutant le code objet d'un programme source.		
La construction de l'arbre abstrait s'effectue après l'analyse syntaxique.		
La table des symboles est une structure indispensable dans l'écriture d'un compilateur.		
Tous les noyaux des états $LR(1)$ correspondent aux états LR(0).		
Je n'ai pas oublié de mettre mon nom sur la feuille. ...		

1. Justifiez le 0 de  $LR(0)$ .

2. Dans le cas de conflits *lecture/réduction*, comment résoudre ces conflits de manière à reconnaître la plus grande chaîne de symboles d'entrée possible?  
Est-ce que cela fonctionne toujours dans le cas de plusieurs conflits *lecture/réduction* sur le même token? (Justifiez).

3. A quoi sert un *ramasse-miettes*? (encore appelé *garbage collector*)

4. Donnez un exemple d'erreur sémantique statique et un exemple d'erreur sémantique dynamique en langage C. Les exemples doivent être "complets" et illustrer clairement le type d'erreur (en particulier, s'il doit y avoir des déclarations, ne les omettez pas).

5. Rappelez l'algorithme de construction d'un item  $LR(1)$ .

6. Donnez la définition du terme *identification*.

7. Qu'appelle-t-on *espace des noms*?

8. Que pouvez-vous dire à propos de l'identification concernant la surcharge en Ada?