

Niveau : 3^{ème} années licence SI
 Module : Compilation

Dimanche 18 janvier 2024
 Durée 1h 30 m

Nom :
 Prénom :
 Groupe :

Examen Final

Exercice 01 (6 points) : choisir les bonnes réponses

- | | |
|--|--|
| <p>1. Si un état dans un automate contient une production finale et une transition donc :</p> <p>a. La grammaire est LR (1) <input type="checkbox"/></p> <p>b. La grammaire est SLR <input type="checkbox"/></p> <p>c. La grammaire n'est pas LR (0) <input checked="" type="checkbox"/></p> <p>d. Aucune de ces réponses. <input type="checkbox"/></p> <p>3. L'analyseur syntaxique qui utilise plusieurs symboles d'anticipation est :</p> <p>a. SLR. <input type="checkbox"/></p> <p>b. LR (1). <input type="checkbox"/></p> <p>c. LR (0). <input type="checkbox"/></p> <p>d. LR (K). <input checked="" type="checkbox"/></p> <p>5. Dans Bison, à quoi sert la règle %% ?</p> <p>a. Définir les unités lexicales <input type="checkbox"/></p> <p>b. Séparer déclarations et règles <input checked="" type="checkbox"/></p> <p>c. Spécifier les actions de réduction <input type="checkbox"/></p> <p>d. Déclarer les variables globales <input type="checkbox"/></p> | <p>2. Lequel des éléments suivants est produit par l'analyseur lexical ?</p> <p>a. Code machine <input type="checkbox"/></p> <p>b. Arbre de syntaxe abstraite (AST) <input type="checkbox"/></p> <p>c. Table des symboles <input checked="" type="checkbox"/></p> <p>d. Code intermédiaire <input type="checkbox"/></p> <p>4. Quelle phase du compilateur traduit le code source en langage machine ?</p> <p>a. Analyse lexicale <input type="checkbox"/></p> <p>b. Analyse syntaxique 1 pt * 6 <input type="checkbox"/></p> <p>c. Génération de code <input checked="" type="checkbox"/></p> <p>d. Optimisation de code <input type="checkbox"/></p> <p>6. Dans Bison, comment définit-on une règle de production pour une expression ?</p> <p>a. $expression : term \mid term "+" expression ;$ <input type="checkbox"/></p> <p>b. $expression = term \mid term "+" expression ;$ <input type="checkbox"/></p> <p>c. $expr : term '+' expr ;$ <input checked="" type="checkbox"/></p> <p>d. $expr \rightarrow term '+' expr ;$ <input type="checkbox"/></p> |
|--|--|

Exercice 02 (4 points) :

$$G1 : \begin{cases} S \rightarrow Sa \mid Ab \\ A \rightarrow Aa \mid Bc \\ B \rightarrow b \mid c \mid \epsilon \end{cases}$$

1. Donnez la suite de dérivation à gauche du mot "bcaba".

$$S \rightarrow Sa \rightarrow Aba \rightarrow Aaba \rightarrow Bcaba \rightarrow bcaba \quad \mathbf{0.5 \text{ pt}}$$
2. Eliminer la récursivité à gauche qui existe dans la grammaire G1.
 Il existe seulement deux récursivités à gauche direct
 - a) $S \rightarrow Sa \mid Ab$
 - b) $A \rightarrow Aa \mid Bc$

Afin d'éliminer ces récursivités on utilise la règle suivante

$$A \rightarrow A\alpha \mid \beta \Rightarrow \begin{cases} A \rightarrow \beta A' \\ A' \rightarrow \alpha A' \mid \epsilon \end{cases}$$

- a) $\begin{cases} S \rightarrow AbS' \\ S' \rightarrow aS' \mid \epsilon \end{cases}$
- b) $\begin{cases} A \rightarrow BcA' \\ A' \rightarrow aA' \mid \epsilon \end{cases}$

La nouvelle grammaire sans récursivité sera : $G1'$: $\begin{cases} S \rightarrow AbS' \\ S' \rightarrow aS' | \epsilon \\ A \rightarrow BcA' \\ A' \rightarrow aA' | \epsilon \\ B \rightarrow b | c | \epsilon \end{cases}$ **0.5 pt**

3. Construire l'analyseur avec la décente récursive pour la grammaire G1

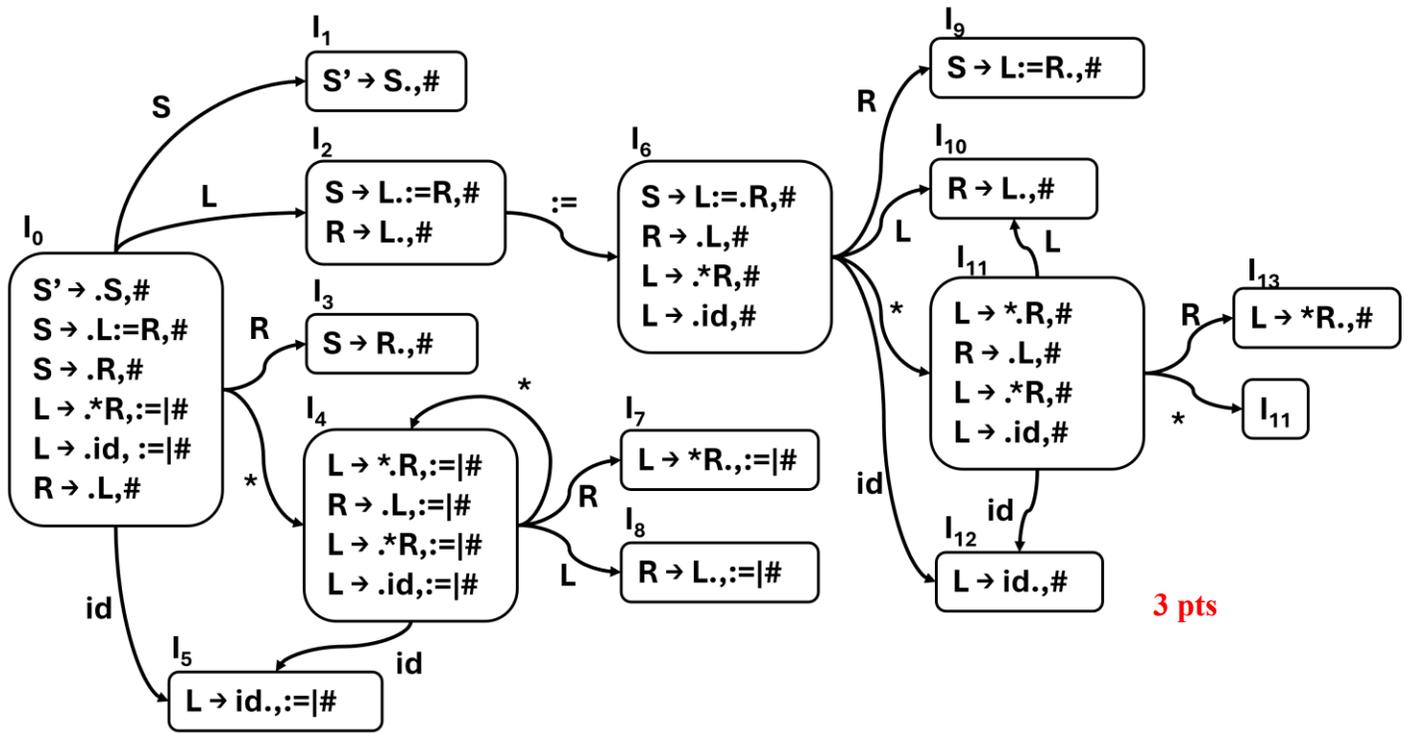
Au Début on ajout l'augmentation à notre grammaire : $E \rightarrow S\#$

<pre> E () { S (); Si(suiv="#") { Accepter (); } Sinon { Erreur (); } } S () { A (); Si(suiv="b") { Suivant (); } Sinon { Erreur (); } } </pre>	<p>0.5 pt * 6</p>	<pre> S' () { Si(suiv="a") { Suivant (); } Sinon { //Rien } } A () { B (); Si(suiv="c") { Suivant (); } Sinon { Erreur (); } } A' () { Si(suiv="a") { Suivant (); } Sinon { //Rien } } B () { Si(suiv="b") { Suivant (); } Sinon { Si(suiv="c") { Suivant (); } Sinon { //Rien } } } </pre>
---	--------------------------	---

Exercice 03 (10 points) :

$$G2 : \begin{cases} S \rightarrow L := R | R \\ L \rightarrow * R | id \\ R \rightarrow L \end{cases}$$

1. Est-ce que G2 est LR (0) ? justifier votre réponse. **Non**, parce qu'il existe dans l'état (I2) une production finale ($R \rightarrow L$.) et une transition avec le symbole « := ». **1 pt**
2. Est-ce que G2 est SLR, justifier votre réponse. **Non**, parce qu'il existe dans l'état 2 une production finale ($R \rightarrow L$.) et une transition avec un des suivant de R (le symbole « := ») **1 pt**
3. Construire l'automate LR (1) pour G2.



3 pts

4. Construire la table d'analyse pour LR (1).

$$G2 : \begin{cases} S \rightarrow L := R & 1 \\ S \rightarrow R & 2 \\ L \rightarrow * R & 3 \\ L \rightarrow id & 4 \\ R \rightarrow L & 5 \end{cases}$$

Etat	Action				Aller à		
	:=	*	id	#	S	L	R
0		S4	S5		1	2	3
1				Acc			
2	S6			R5			
3				R2			
4		S4	S5			8	7
5	R4			R4			
6		S11	S12			10	9
7	R3			R3			
8	R5			R5			
9				R1			
10				R5			
11		S11	S12			10	13
12				R4			
13				R3			

3 pts

5. À partir de l'automate LR (1) construit, Quelles sont les états à fusionner pour créer l'analyseur LALR(1) ?

Les états sont :

- a) Etat 8 et état 10
- b) Etat 4 et état 11
- c) Etat 5 et état 12
- d) Etat 7 et état 13

1 pt

6. Analyser le mot « id := id ».

Pile	Entrée	Action
0	id :=id#	S (id), état = 5
0id5	:=id#	D (2), E (L)
0L	:=id#	Etat = 2
0L2	:=id#	S (:=), etat = 6
0L2 :=6	id#	S (id), etat = 12
0L2 :=6id12	#	D (2), E(L)
0L2 :=6L	#	Etat = 10
0L2 :=6L10	#	D (2), E (R)
0L2 :=6R	#	Etat = 9
0L2 :=6R9	#	D (6), E (S)
0S	#	Etat = 1
0S1	#	Accepter

1 pt