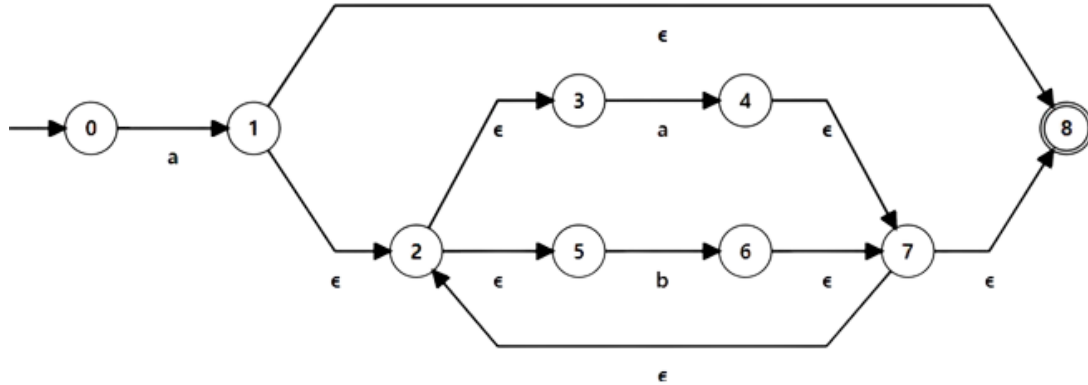


Corrigé type de l'examen n ° 01 de compilation

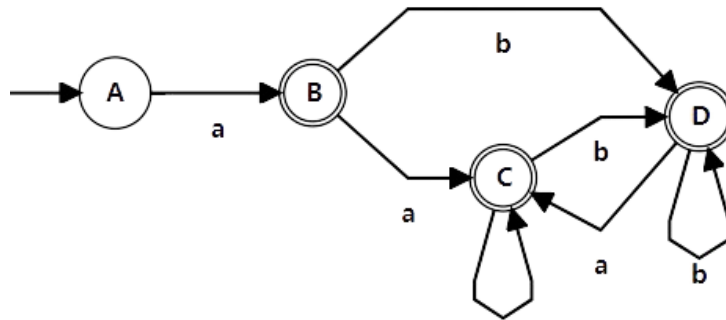
EXERCICE 1. (7 points)

1. AFN de l'expression régulière suivante $a(a|b)^*$, en utilisant les règles de Thompson : (2pts)

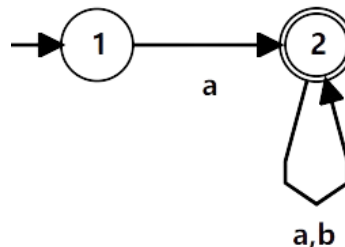


2. Rendre l'automate non déterministe (AFN) de la question précédente déterministe (AFD). (3pts)

Etats AFN	Etats AFD	a	b	Type
{0}	A	B		
{1, 2, 3, 5, 8}	B	C	D	Final
{2, 3, 4, 5, 7, 8}	C	C	D	Final
{2, 3, 5, 6, 7, 8}	D	C	D	Final



3. AFD pour l'expression régulière précédente sans passer par la construction de Thompson. (1pt)



4. Comparer l'automate proposé dans la question (3) avec l'automate de la question (2) en matière de nombre d'états. (1pts)

L'automate de la question 3 contient moins d'états (2 états) par rapport à l'automate de la question 2 (4 états). L'automate de la question 2 ne représente pas l'automate minimal.

EXERCICE 2. (14 points)

Soit la grammaire G des expressions postfixes :

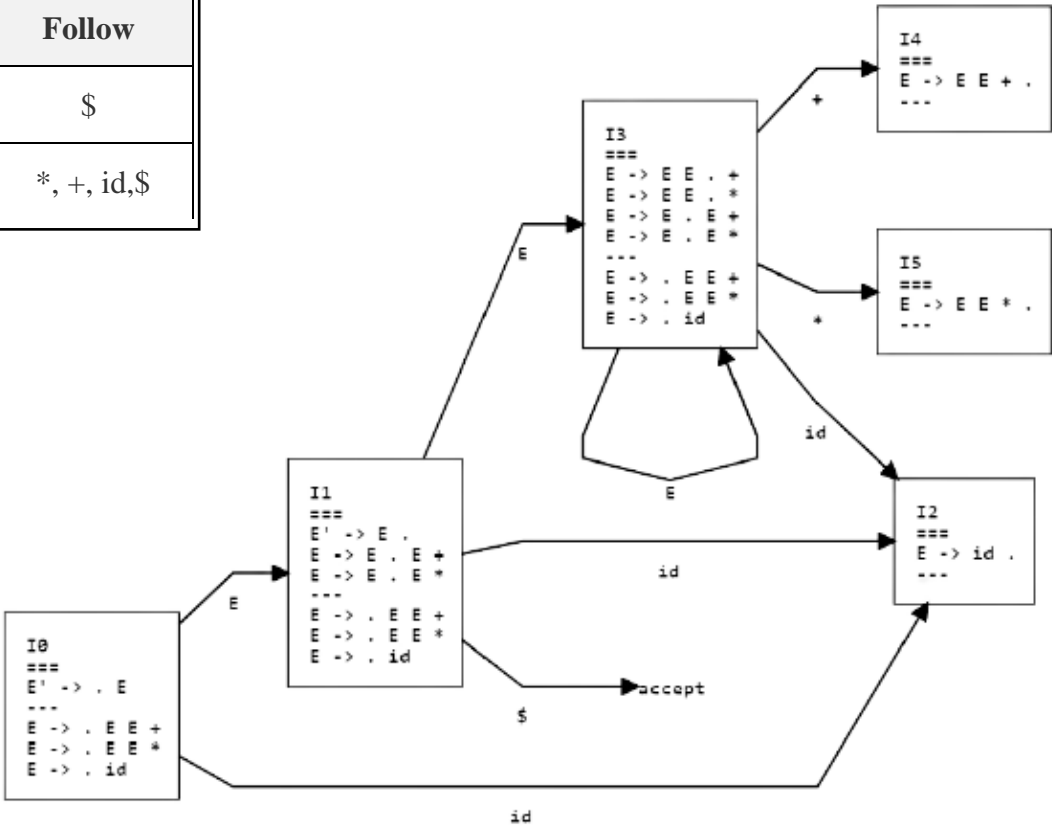
$G : E \rightarrow EE +$

$E \rightarrow EE *$

$E \rightarrow id$

1. Créer la table d'analyse SLR (1) pour la grammaire G. (4pts)

Symbol	First	Follow
S	id	\$
E	id	*, +, id, \$



Etat	ACTION				Aller_à
	*	+	id	\$	
0			s2		1
1			s2	Accepter	3
2	$r(E \rightarrow id)$	$r(E \rightarrow id)$	$r(E \rightarrow id)$	$r(E \rightarrow id)$	
3	s5	s4	s2		3
4	$r(E \rightarrow EE +)$	$r(E \rightarrow EE +)$	$r(E \rightarrow EE +)$	$r(E \rightarrow EE +)$	
5	$r(E \rightarrow EE *)$	$r(E \rightarrow EE *)$	$r(E \rightarrow EE *)$	$r(E \rightarrow EE *)$	

2. Est-ce que la grammaire G est ambiguë ? (1pts)

La grammaire est SLR (1) car la table SLR ne présente aucun conflit → La grammaire n'est pas ambiguë.

3. Est-ce que la grammaire G est LL(1) ? (1pts)

La grammaire n'est pas LL(1) car elle contient la récursivité à gauche.

4. Factoriser à gauche la grammaire G. (1.5pts)

$E \rightarrow E E A \mid id$

$A \rightarrow + \mid *$

5. Éliminer la récursivité gauche dans la grammaire produite en question 4. (1.5pts)

$E \rightarrow id B$

$A \rightarrow + \mid *$

$B \rightarrow E A B \mid \epsilon$

6. Créer la table d'analyse LL(1) pour la grammaire produite en question 5. (4pts)

$S \rightarrow E$

$E \rightarrow id B$

$A \rightarrow + \mid *$

$B \rightarrow E A B \mid \epsilon$

Non terminal	Début	Suivant
S	<i>id</i>	\$
E	<i>id</i>	+,*, \$
A	+, *	<i>id</i>, +, *, \$
B	<i>id</i>, ϵ	+, *, \$

Non terminal	+	*	<i>id</i>	\$
S			$S \rightarrow E\\$	
E			$E \rightarrow id B$	
A	$A \rightarrow +$	$A \rightarrow *$		
B	$B \rightarrow \epsilon$	$B \rightarrow \epsilon$	$B \rightarrow E A B$	$B \rightarrow \epsilon$

⇒ **La grammaire est LL(1) car il n'existe aucun conflit dans la table d'analyse LL(1)**

7. Selon cet exercice, quelle est la méthode la plus puissante entre SLR(1) et LL(1) ? justifier votre réponse. (1pt)

SLR(1) est plus puissant que LL(1) car :

- a. G est directement SLR(1) sans utiliser aucune conversions.**
- b. G nécessite des conversions (Factorisation + élimination de la récursivité gauche) pour devenir LL(1)**