

Contrôle

Fait le : 14/01/2020

Durée : 1h30m

Exercice 01 (06 pts) :

1. Donner une expression régulière du langage formé des mots de longueur au moins 2 sur $\{a, b\}$ pour lesquels tous les a éventuellement présents précèdent les b (éventuellement présents) (**01.5 pts**) .
2. Donner une expression régulière du langage formé des mots sur $\{a,b,c\}$ qui ne contiennent pas deux a consécutifs. (**01.5 pts**) .
3. En java, le type double permet de représenter les nombres à virgule flottante. Un nombre double peut contenir un point '.' (symbolisant la virgule) mais ce n'est pas obligatoire. Lorsqu'il possède un point, il y a forcément un chiffre avant ou après le point. Un littéral double peut posséder un signe + ou -. Il peut aussi posséder un facteur d'échelle qui est représenté par nombre entier signé précédé par (e ou E, pour exponentiel). Voici des exemples de nombres double :

3.86473e5 123.0e+8 -12E4 .45E-23 -16 012.e78

En utilisant Lex, écrire un programme reconnaissant les nombres double de java (**03 pts**).

Exercice 02 (08 pts) :

Soit G la grammaire suivante :

$S \rightarrow BAc \mid a$

$A \rightarrow bSBa \mid \varepsilon$

$B \rightarrow dBcS \mid \varepsilon$

1. Construire les ensembles PREMIER et SUIVANT des symboles de G .
2. G est-elle $LL(1)$?
3. Si oui analyser la chaîne « **dcbaacc** ».
4. Transformer G en une grammaire G' ε -libre équivalente. G' est-elle $LL(1)$?

Exercice 03 (06 pts) :

Soit la grammaire G d'axiome S , de terminaux $\{a, b, d, e\}$ et de productions:

$S \rightarrow AB \mid Da$

$A \rightarrow aAb \mid \varepsilon$

$B \rightarrow bB \mid \varepsilon$

$D \rightarrow dD \mid e$

1. Construire l'automate LR de cette grammaire.
2. Est-ce que G est de type SLR(1)? Justifier rigoureusement en énumérant tous les conflits et en précisant leur type (shift/reduce ou reduce/reduce)..
3. Est-ce que G est de type LR(0)? Justifiez votre réponse sans construire la table d'analyse.

Corrigé type

Fait le : 14/01/2020

Durée : 1h30m

Exercice 01 (06 pts) :

1. $aa^*bb^*|aaa^*|bbb^*$ (01.5 pts) .
2. $(ac|ab|b|c)(ac|ab|b|c)^*(a|\epsilon)$ (01.5 pts) .
3. le type double en java (03 pts)

```
#include<stdio.h>
```

```
chiffre [0-9]
```

```
nbr {chiffre}+
```

```
sig [\+|-]
```

```
Exp [Ee]{sig}?{nbr}
```

```
nbr1 {nbr}(\.){nbr}?{Exp}?
```

```
nbr2 {nbr}?(\.){nbr}){Exp}?
```

```
doubleJava {sig}?({nbr1}|{ nbr2})
```

```
%%
```

```
{ doubleJava } return {double}
```

```
%%
```

Exercice 02 (08 pts) :

Soit G la grammaire suivante :

$S \rightarrow BAc | a$

$A \rightarrow bSBa | \epsilon$

$B \rightarrow dBcS | \epsilon$

5. Construire les ensembles PREMIER et SUIVANT des symboles de G. (02 pts).

	Premiers	Suivants
S	a,b,c, d	a, d, c, \$
A	b, ϵ	c
B	d, ϵ	a, b, c

6. G est-elle LL(1) (02 pts). ?

	a	b	c	d	\$
S	a	Bac	BAc	Bac	
A		bSBa	ϵ		
B	ϵ	ϵ	ϵ	dBcS	

7. Si oui analyser la chaîne « dcbaacc » (02 pts)..

Chaîne	Pile	Action
dcbaacc\$	\$S	dépiler, empiler cAB
dcbaacc\$	\$cAB	dépiler, empiler ScBd
dcbaacc\$	\$cAScBd	dépiler, Avancer
cbaacc\$	\$cAScB	dépiler, empiler ϵ
cbaacc\$	\$cASc	dépiler, Avancer

baacc\$	\$cAS	dépiler, empiler cAB
baacc\$	\$cAcAB	dépiler, empiler ε
baacc\$	\$cAcA	dépiler, empiler aBSb
baacc\$	\$cAcaBSb	dépiler, Avancer
aacc\$	\$cAcaBS	dépiler, empiler a
aacc\$	\$cAcaBa	dépiler, Avancer
acc\$	\$cAcaB	dépiler, empiler ε
acc\$	\$cAca	dépiler, Avancer
cc\$	\$cAc	dépiler, Avancer
c\$	\$cA	dépiler, empiler ε
c\$	\$c	dépiler, Avancer
\$	\$	Accépter

8. Transformer G en une grammaire G' ε -libre équivalente (02 pts)..

Sol1 : $S \rightarrow BAc \mid Bc \mid Ac \mid a$

$A \rightarrow bSBa \mid bSa$

$B \rightarrow dBcS \mid dcS$

G' n'est pas LL(1)

Sol2 : $S \rightarrow BS' \mid Ac \mid a$

$S' \rightarrow Ac \mid c$

$A \rightarrow bSA'$

$A' \rightarrow Ba \mid bSa$

$B \rightarrow dB'$

$B' \rightarrow BcS \mid cS$

G' est LL(1)

Exercice 03 (06 pts) :

Soit la grammaire G d'axiome S, de terminaux {a, b, d, e} et de productions :

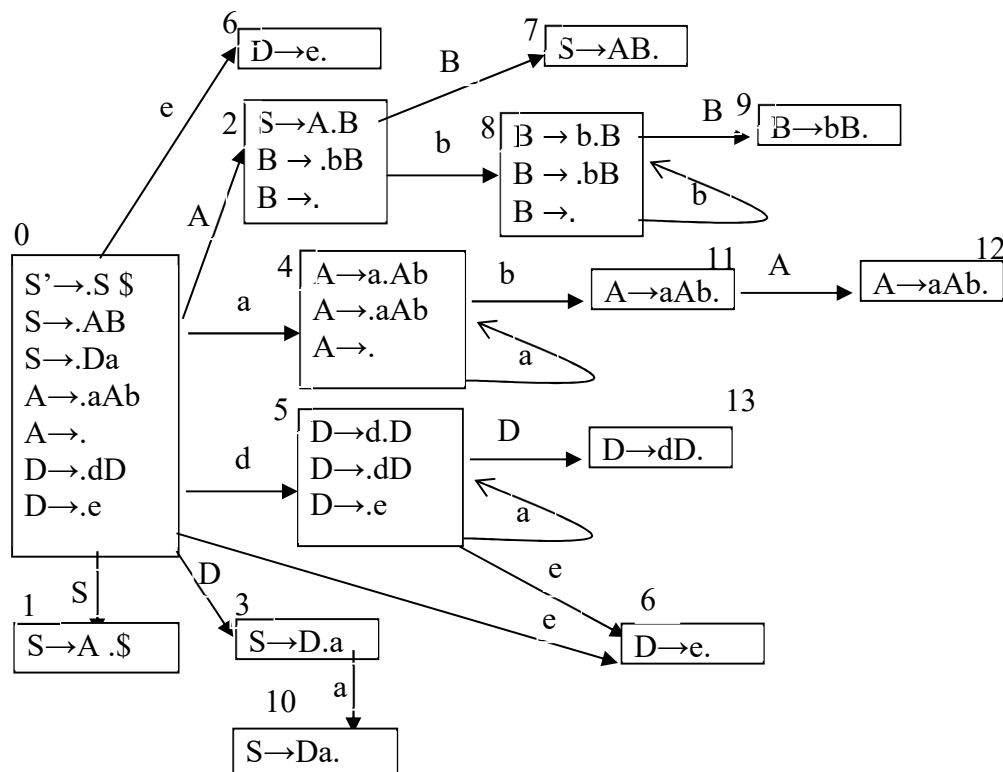
$S \rightarrow AB \mid Da$

$A \rightarrow aAb \mid \epsilon$

$B \rightarrow bB \mid \epsilon$

$D \rightarrow dD \mid e$

1. Construire l'automate LR de cette grammaire. (02 pts)



2. Est-ce que G est de type SLR(1)? Justifier rigoureusement en énumérant tous les conflits et en précisant leur type (shift/reduce ou reduce/reduce).. **(02 pts)**

SLR(1)	a	b	d	e	\$		S	A	B	D
0	d4	r5	d5	d6	r5		1	2		3
1					Acc					
2		d8			r6				7	
3	d10									
4	d4	r4			r4					
5			d5	d6						13
6	r8									
7					r1					
8		d8			r6				9	
9					r5					
10					r2					
11		d12								
12		r3			r3					
13	r7									

(01 pts)	Premiers	Suivants
S	a,b,e,d	\$
A	a,ε	\$,b
B	b,ε	\$
D	d,e	a

3. Est-ce que G est de type LR(0)? Justifiez votre réponse sans construire la table d'analyse.

N'est pas LR(0) puisque il existe plusieurs cas de conflits par exemple dans l'état 0etc. **(01 pts)**

Rattrapage

Date : 18/02/2020

Durée : 1h30m

Exercice 01 (06 pts)

1. Écrire une expression régulière qui définit un nombre décimal avec virgule fixe sans zéros au début ni à la fin.
Exemples :
00.0 **123.01** **1230005.0** sont acceptées,
00.0 **001.000** **0002345.1000** ne sont pas acceptées.
2. Donner une expression régulière définissant le langage composé de tous les mots non vides, sur l'alphabet $\{0,1\}$, qui contient au moins deux **1** mais pas des **1** consécutifs.
3. Donner une expression régulière définissant le langage composé de tous les mots non vides, sur l'alphabet $\{0,1\}$, qui contient au plus deux **1** mais pas des **1** consécutifs.
4. Donner une expression régulière définissant le langage composé de tous les mots non vides, sur les chiffres décimaux dont ces mots ne dépassent pas le nombre **2020**.

Exercice 02 (06 pts)

Soit G la grammaire suivante :

$$\begin{aligned} S &\rightarrow Aa|Bb \\ A &\rightarrow aA|b \\ B &\rightarrow bc|bD \\ D &\rightarrow De|fFg|fe \\ F &\rightarrow fDg \end{aligned}$$

1. Cette grammaire n'est pas LL(1) : pourquoi ?
2. Donner une grammaire G1 équivalente à G et qui vous semble LL(1).
3. Donner la table d'analyse LL(1) de G1. Justifier en utilisant cette table que G1 est une grammaire LL(1).

Exercice 03 (08 pts)

Soit la grammaire $G = (VN, VT, T, P)$ avec:

$VT = \{., =, \text{key}, \text{nl}, \text{str}, \text{num}\},$
 $VN = \{T, E, F, K, L, V\}$
et P est l'ensemble de règles :

$$\begin{aligned} T &\rightarrow E F \\ F &\rightarrow \text{nl } E F \mid \varepsilon \\ E &\rightarrow K = V \\ K &\rightarrow \text{key } L \\ L &\rightarrow . \text{key } L \mid \varepsilon \\ V &\rightarrow \text{str} \mid \text{num} \end{aligned}$$

1. Construire l'automate LR(0) pour G.
2. Au vu de l'automate, exhibez tous les états pour lesquels il y aura un conflit dans la table Action de l'automate LR(0), et explicitez ces conflits.
3. Construire la table de l'analyseur SLR(1) pour G. G est-elle SLR(1) ? Justifiez.
4. Effectuez la reconnaissance du mot **key = str nl key.key = num** à l'aide de table SLR(1). En cas de conflit, précisez l'action choisie.

Corrigé type du Rattrapage

Fait le : 04/03/2014

Durée : 1h30m

Exercice 01 (06 pts)

1. Écrire une expression régulière qui définit un nombre décimal avec virgule fixe sans zéros au début ni à la fin.
 $0([1-9]+0^*)(\.(0*[1-9])^+)$
2. Donner une expression régulière définissant le langage composé de tous les mots non vides, sur l'alphabet $\{0,1\}$, qui contient au moins deux 1 mais pas des 1 consécutifs.
 $(00^*100^*1)^+(0)^*|(100^*100^*)^+(1|\epsilon)$
3. Donner une expression régulière définissant le langage composé de tous les mots non vides, sur l'alphabet $\{0,1\}$, qui contient au plus deux 1 mais pas des 1 consécutifs.
 $1|0^*1?00^*1?0^*$
4. Donner une expression régulière définissant le langage composé de tous les mots non vides, sur les chiffres décimaux dont ces mots ne dépassent pas le nombre 2020.
 $[0-1]?[0-9]\{3\}|20([0-1][0-9]|20)$

Exercice 02 (06 pts)

Exercice 03 (08 pts) :