

TD03 - ANALYSE SLR

L'objectif de ce TD est de construire des analyseurs SLR à partir des grammaires suivantes :

G_1 : parenthèses équilibrées

- (1) $P \rightarrow (P) P$
- (2) $P \rightarrow \varepsilon$

G_2 : déclarations simplifiées en L , \mathbf{dv} = déclaration de variable, \mathbf{df} = déclaration de fonction

- (1) $P \rightarrow V ; F$
- (2) $V \rightarrow \mathbf{dv} V'$
- (3) $V' \rightarrow , \mathbf{dv} V'$
- (4) $V' \rightarrow \varepsilon$
- (5) $F \rightarrow \mathbf{df} F$
- (6) $F \rightarrow \varepsilon$

Suivez les étapes ci-dessous pour chacune des grammaires G_1 et G_2 .

Exercice 1. *Augmentation de la grammaire*

Créez une grammaire augmentée G' en rajoutant un nouvel axiome S dont la seule production est $S \rightarrow P$ vers l'ancien axiome P de la grammaire originale G .

Correction:

G'_1 :

- (0) $S \rightarrow P$
- (1) $P \rightarrow (P) P$
- (2) $P \rightarrow \varepsilon$

G'_2 :

- (0) $S \rightarrow P$
- (1) $P \rightarrow V ; F$
- (2) $V \rightarrow \mathbf{dv} V'$
- (3) $V' \rightarrow , \mathbf{dv} V'$
- (4) $V' \rightarrow \varepsilon$
- (5) $F \rightarrow \mathbf{df} F$
- (6) $F \rightarrow \varepsilon$

Exercice 2. *Articles/FERMETURE*

Un *article* est une production avec un marqueur spécial \bullet en partie droite de la production. Ce marqueur indique la partie du manche qui a déjà été observée sur la pile.

Construisez l'ensemble d'articles initial $\text{FERMETURE}(\{S \rightarrow \bullet P\})$. La fonction $\text{FERMETURE}(I)$ prend un ensemble d'articles et, pour tout article $A \rightarrow \alpha \bullet B \gamma$ dans $\text{FERMETURE}(I)$, ajoute $B \rightarrow \bullet \beta$ pour toute règle $B \rightarrow \beta$ de la grammaire

Correction:

$G'_1 : I_0 = \{S \rightarrow \bullet P, P \rightarrow \bullet, P \rightarrow \bullet (P) P\}$

$G'_2 : I_0 = \{S \rightarrow \bullet P, P \rightarrow \bullet V ; F, V \rightarrow \bullet \mathbf{dv} V'\}$

Exercice 3. *Fonction ALLER_A(I, X)*

À partir d'un ensemble d'articles I , et d'un symbole X terminal ou non terminal de la grammaire, la fonction $\text{ALLER_A}(I, X)$ est la FERMETURE de l'ensemble de tous les articles $A \rightarrow \alpha X \bullet \beta$ tels que $A \rightarrow \alpha \bullet X \beta$ est dans I . C'est-à-dire, la fonction $\text{ALLER_A}(I, X)$ fait avancer le marqueur \bullet pour tous les articles de I dont le prochain symbole après le marqueur est X .

Calculez la collection d'ensembles d'articles de la grammaire à l'aide de la fonction $\text{ALLER_A}(I, X)$ à partir de l'ensemble initial déjà calculé, pour tous les symboles X après le marqueur \bullet . Itérez ce processus jusqu'à ce qu'aucun nouvel ensemble d'articles ne soit ajouté à la collection.

Correction:

$G'_1 :$

$$\begin{aligned} I_0 &= \{S \rightarrow \bullet P, P \rightarrow \bullet, P \rightarrow \bullet (P) P\} \\ I_1 &= \{S \rightarrow P \bullet\} \\ I_2 &= \{P \rightarrow (\bullet P) P, P \rightarrow \bullet, P \rightarrow \bullet (P) P\} \\ I_3 &= \{P \rightarrow (P \bullet) P\} \\ I_4 &= \{P \rightarrow (P) \bullet P, P \rightarrow \bullet, P \rightarrow \bullet (P) P\} \\ I_5 &= \{P \rightarrow (P) P \bullet\} \end{aligned}$$

$G'_2 :$

$$\begin{aligned} I_0 &= \{S \rightarrow \bullet P, P \rightarrow \bullet V ; F, V \rightarrow \bullet \text{ dv } V'\} \\ I_1 &= \{S \rightarrow P \bullet\} \\ I_2 &= \{V \rightarrow \text{ dv } \bullet V', V' \rightarrow \bullet, \text{ dv } V', V' \rightarrow \bullet\} \\ I_3 &= \{P \rightarrow V \bullet ; F\} \\ I_4 &= \{V \rightarrow \text{ dv } V' \bullet\} \\ I_5 &= \{V' \rightarrow , \bullet \text{ dv } V'\} \\ I_6 &= \{P \rightarrow V ; \bullet F, F \rightarrow \bullet \text{ df } F, F \rightarrow \bullet\} \\ I_7 &= \{V' \rightarrow , \text{ dv } \bullet V', V' \rightarrow \bullet, \text{ dv } V', V' \rightarrow \bullet\} \\ I_8 &= \{V' \rightarrow , \text{ dv } V' \bullet\} \\ I_9 &= \{P \rightarrow V ; F \bullet\} \\ I_{10} &= \{F \rightarrow \text{ df } \bullet F, F \rightarrow \bullet \text{ df } F, F \rightarrow \bullet\} \\ I_{11} &= \{F \rightarrow \text{ df } F \bullet\} \end{aligned}$$

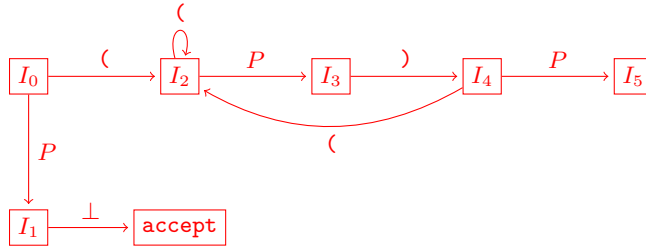
Exercice 4. Automate LR(0)

Dessinez l'automate LR(0) de la grammaire de la façon suivante :

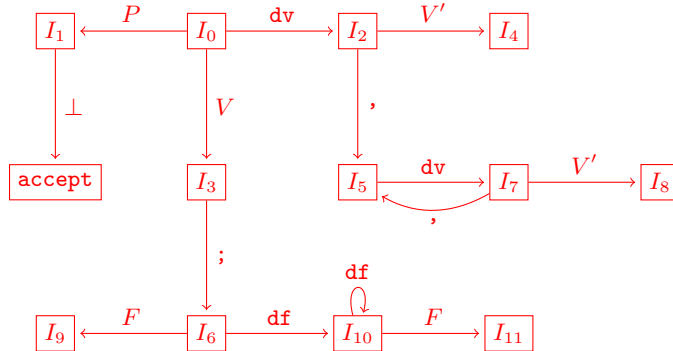
- Les états sont les ensembles d'articles I calculés précédemment
- Les transitions sont données par les valeurs de ALLER_A(I, X)
- L'état initial est FERMETURE($\{S \rightarrow \bullet P\}$)
- Une transition étiquetée \$ depuis l'état FERMETURE($\{S \rightarrow P \bullet\}$) mène à l'état **accept**

Correction:

$G'_1 :$



$G'_2 :$



Exercice 5. PREMIER(X)/SUIVANT(X)

Pour chaque symbole non terminal X de la grammaire, calculez PREMIER(X) et SUIVANT(X). Pour calculer PREMIER(X), appliquer les règles suivantes jusqu'à ce qu'aucun symbole ne puisse être ajouté à PREMIER(X).

- (1) Si $X \rightarrow \varepsilon$ est une production de la grammaire, on ajoute ε à PREMIER(X).

- (2) Si $X \rightarrow Y_1 \dots Y_k \in P$, mettre a dans $\text{PREMIER}(X)$ s'il existe i tel que a est dans $\text{PREMIER}(Y_i)$ et que ε est dans tous les $\text{PREMIER}(Y_1) \dots \text{PREMIER}(Y_{i-1})$. Si $\varepsilon \in \text{PREMIER}(Y_j) \forall j, 1 \leq j \leq k$, on ajoute ε à $\text{PREMIER}(X)$.¹

Pour calculer $\text{SUIVANT}(X)$, appliquer les règles suivantes jusqu'à ce qu'aucun symbole ne puisse être ajouté à $\text{SUIVANT}(X)$.

- (1) Mettre \perp dans $\text{SUIVANT}(S)$.
- (2) si $X \rightarrow \alpha B \beta$, le contenu de $\text{PREMIER}(\beta)$, excepté ε , est ajouté à $\text{SUIVANT}(B)$.
- (3) s'il existe une règle $X \rightarrow \alpha B$ ou une règle $X \rightarrow \alpha B \beta$ telle que $\varepsilon \in \text{PREMIER}(\beta)$ (c'est à dire $\beta \Rightarrow^* \varepsilon$), les éléments de $\text{SUIVANT}(X)$ sont ajoutés à $\text{SUIVANT}(B)$.

Correction:

$G'_1 :$

X	PREMIER(X)	SUIVANT(X)
S	(ε	\perp
P	(ε	\perp)

$G'_2 :$

X	PREMIER(X)	SUIVANT(X)
S	dv	\perp
P	dv	\perp
V	dv	;
V'	, ε	;
F	df ε	\perp

Exercice 6. Construction de la table SLR

Construisez la table d'analyse SLR pour la grammaire comme suit.

- (1) Construire $C = \{I_0, I_1, \dots, I_n\}$ la collection d'ensemble d'articles $LR(0)$ pour G'
- (2) L'état i est construit à partir de I_i . Les actions d'analyse syntaxique pour l'état i sont déterminées comme suit :
 - (a) Si $A \rightarrow \alpha \bullet a \beta$ est dans I_i et si $\text{ALLER_A}(I_i, a) = I_j$, alors $\text{ACTION}[i, a] = dj$. Dans ce cas, a doit être un terminal.
 - (b) Si $A \rightarrow \alpha \bullet$ est dans I_i , alors $\text{ACTION}[i, a] = rj$ où j est le numéro de la règle $A \rightarrow \alpha$ pour tout $a \in \text{SUIVANT}(A)$, à l'exception de S'
 - (c) Si $S' \rightarrow S \bullet$ est dans I_i , alors $\text{ACTION}[i, \$] = acc$

Si un conflit entre différentes actions résulte de ces règles, la grammaire n'est pas SLR.

- (3) Les transitions de transfert $\text{GOTO}[i, A]$ pour l'état i sont construites pour tout non terminal A comme suit : si $\text{ALLER_A}(I_i, A) = I_j$ alors $\text{GOTO}[i, A] = j$
- (4) Toutes les entrées non remplies par les règles 2 et 3 sont positionnées à **err** (cellules vides)
- (5) L'état initial est celui construit à partir de l'ensemble d'items contenant $S' \rightarrow \bullet S$

Correction:

$G'_1 :$

	ACTION[,]			GOTO[,]
	()	\perp	P
0	d2	r2	r2	1
1			acc	
2	d2	r2	r2	3
3		d4		
4	d2	r2	r2	5
5		r1	r1	

$G'_2 :$

	ACTION[,]					GOTO[,]			
	dv	df	,	;	\perp	P	V	V'	F
0	d2					1	3		
1					acc				
2			d5	r4				4	
3				d6					
4				r2					
5	d7								
6		d10			r6				9
7			d5	r4				8	
8				r3					
9					r1				
10		d10			r6				11
11					r5				

1. $\text{PREMIER}(a) = \{a\}$ pour a terminal.

Exercice 7. Analyse LR

L'analyseur se sert de la table SLR pour déterminer la prochaine action en fonction de l'état au sommet de la pile i et du prochain symbole sous la tête de lecture a de la façon suivante :

- Si $\text{ACTION}[i, a] = dj$, où j est un état. L'analyseur effectue un décalage : il empile j et consomme une unité lexicale
- Si $\text{ACTION}[i, a] = rk$, où k est le numéro de la règle $A \rightarrow \beta$. L'analyseur effectue une réduction :
 - il dépile $|\beta|$ symboles de la pile
 - l'état l est maintenant au sommet de la pile
 - il empile l'état m , qui correspond à l'entrée $\text{GOTO}[l, A]$
- Si $\text{ACTION}[i, a] = \text{acc}$: l'analyseur accepte l'entrée
- Si $\text{ACTION}[i, a] = \text{err}$: l'analyseur signale une erreur

Utilisez la table d'analyse pour simuler l'analyse des mots suivants :

- Pour $G_1 : w_1 = (())$ et $w_2 = ()$
- Pour $G_2 : w_1 = \text{dv} ; \text{df}$, $w_2 = \text{dv} ; \text{dv}$ et $w_3 = \text{dv} , \text{dv} ;$

Détaillez l'état de la pile, de la bande de lecture et l'action de l'analyseur à chaque étape.

Correction:

Format (pile,buffer,action) avec le sommet de la pile à droite.

$G_1, w_1 = (()) :$

(\perp 0, (()) \perp , d2)

(\perp 0 2, ()) \perp , d2)

(\perp 0 2 2,)) \perp , r2)

(\perp 0 2 2 3,)) \perp , d4)

(\perp 0 2 2 3 4,) \perp , r2)

(\perp 0 2 2 3 4 5,) \perp , r1)

(\perp 0 2 3,) \perp , d4)

(\perp 0 2 3 4, \perp , r2)

(\perp 0 2 3 4 5, \perp , r1)

(\perp 0 1, \perp , acc)

$G_1, w_2 = () :$

(\perp 0, ()) \perp , d2)

(\perp 0 2,)) \perp , r2)

(\perp 0 2 3,)) \perp , d4)

(\perp 0 2 3 4,) \perp , r2)

(\perp 0 2 3 4 5,) \perp , r1)

(\perp 0 1,) \perp , err)

$G_2, w_1 = \text{dv} ; \text{df} :$

(\perp 0, $\text{dv} ; \text{df}$ \perp , d2)

(\perp 0 2, ; df \perp , r4)

(\perp 0 2 4, ; df \perp , r2)

(\perp 0 3, ; df \perp , d6)

(\perp 0 3 6, df \perp , d10)

(\perp 0 3 6 10, \perp , r6)

(\perp 0 3 6 10 11, \perp , r5)

(\perp 0 3 6 9, \perp , r1)

(\perp 0 1, \perp , acc)

$G_2, w_2 = \text{dv} ; \text{dv} :$

(\perp 0, $\text{dv} ; \text{dv}$ \perp , d2)

(\perp 0 2, ; dv \perp , r4)

(\perp 0 2 4, ; dv \perp , r2)

(\perp 0 3, ; dv \perp , d6)

(\perp 0 3 6, dv \perp , err)

$G_2, w_3 = \text{dv} , \text{dv} ; :$

(\perp 0, $\text{dv} , \text{dv} ; \perp$, d2)

(\perp 0 2, , $\text{dv} ; \perp$, d5)

(\perp 0 2 5, $\text{dv} ; \perp$, d7)

(\perp 0 2 5 7, ; \perp , r4)

(\perp 0 2 5 7 8, ; \perp , r3)

(\perp 0 2 4, ; \perp , r2)

(\perp 0 3, ; \perp , d6)

(\perp 0 3 6, \perp , r6)

(\perp 0 3 6 9, \perp , r1)

(\perp 0 1, \perp , acc)