

# Théorie des langages : THL

## CM 5

Uli Fahrenberg

EPITA Rennes

S5 2021

# Aperçu

# Programme du cours

- ① Langages rationnels
- ② Automates finis
- ③ Langages algébriques, grammaires hors-contexte, automates à pile
- ④ Parsage LL, partie 1
- ⑤ **Parsage LL, partie 2**
- ⑥ TP 1 : flex
- ⑦ Parsage LR
- ⑧ TP 2, 3 : flex & bison

QCM 2

# La dernière fois : parsing

## Problème de parsing

Pour une grammaire hors contexte  $G$ , construire un **algorithme de parsing** qui

- pour un mot  $w$ , décide si  $w \in L(G)$ ,
- et dans le cas  $w \in L(G)$ , retourne l'arbre de dérivation.

- arbre de dérivation de  $w \hat{=}$  **sémantique** de  $w$

Nos algorithmes de parsing devraient

- pouvoir traiter des grammaires non-ambiguës
- avoir une complexité linéaire en taille d'entrée
- lire  $w$  de gauche à droite sans retour arrière

# Language Trouble

anglais	français	Uli	
parsing	analyse syntaxique	parsage	
parse tree	arbre d'analyse	arbre de parsage	

# Language Trouble

anglais	français	Uli	Fabrice
parsing	analyse syntaxique	parsage	parsing
parse tree	arbre d'analyse	arbre de parsage	AST

Abstract Syntax Tree

## La dernière fois : parsage LL(1)

- approche descendante
- lire le mot  $w$  de gauche à droite / Left-to-right
  - sans passer à l'arrière
- construire une dérivation gauche / Leftmost
- en accordant, à chaque pas, le premier symbole de  $w$  avec le côté droit d'une production
  - donc avec lookahead 1
- parsage LL( $k$ ) : lookahead  $k$  / « fenêtre de  $k$  lexèmes »
- peu utilisé

# La dernière fois : algorithme LL(1)

- ① entrée : une grammaire hors contexte  $G = (N, \Sigma, P, S)$ 
  - si-dessous,  $V = N \cup \Sigma$
  - éliminer récursion à gauche dans  $G$  ; factoriser  $G$  à gauche
- ② calculer NULL
  - $NULL = \{A \in N \mid A \Rightarrow^* \varepsilon\}$
- ③ construire la table FIRST
  - $FIRST(A) = \{a \in \Sigma \mid \exists w \in V^* : A \Rightarrow^* aw\}$
- ④ construire la table FOLLOW
  - $FOLLOW(A) = \{a \in \Sigma \mid \exists B \in N, \alpha, \beta \in V^* : B \Rightarrow^* \alpha A a \beta\}$
- ⑤ construire la TABLE de passage :
  - ① pour chaque production  $X \rightarrow w$  ( $n$ ) :
    - ① pour chaque  $a \in FIRST(w)$  :  $TABLE(X, a) += \{n\}$
    - ② si  $w \in NULL$  ou  $w = \varepsilon$  :
      - pour chaque  $a \in FOLLOW(X)$  :  $TABLE(X, a) += \{n\}$



# La dernière fois : LL(1)

## Définition (8.5)

$G$  est **LL(1)** si chaque  $\text{TABLE}(A, a)$  contient **au maximum une** production.

# La dernière fois : exemple

$$Z \rightarrow XYZ \quad (1)$$

$$| c \quad (2)$$

$$X \rightarrow a \quad (3)$$

$$| Y \quad (4)$$

$$Y \rightarrow b \quad (5)$$

$$| \varepsilon \quad (6)$$

# La dernière fois : exemple

$$Z \rightarrow XYZ \quad (1)$$

$$| c \quad (2)$$

$$X \rightarrow a \quad (3)$$

$$| Y \quad (4)$$

$$Y \rightarrow b \quad (5)$$

$$| \varepsilon \quad (6)$$

$$\text{NULL} = \{X, Y\}$$

# La dernière fois : exemple

$$Z \rightarrow XYZ \quad (1)$$

$$\quad | c \quad (2)$$

$$X \rightarrow a \quad (3)$$

$$\quad | Y \quad (4)$$

$$Y \rightarrow b \quad (5)$$

$$\quad | \varepsilon \quad (6)$$

$$\text{NULL} = \{X, Y\}$$

$A$	$\text{FIRST}(A)$
$X$	$a, b$
$Y$	$b$
$Z$	$c, a, b$

# La dernière fois : exemple

$$Z \rightarrow XYZ \quad (1)$$

$$\quad | c \quad (2)$$

$$X \rightarrow a \quad (3)$$

$$\quad | Y \quad (4)$$

$$Y \rightarrow b \quad (5)$$

$$\quad | \varepsilon \quad (6)$$

$$\text{NULL} = \{X, Y\}$$

A	FIRST(A)
X	a, b
Y	b
Z	c, a, b

A	FOLLOW(A)
X	a, b, c
Y	a, b, c
Z	

## La dernière fois : exemple

$$Z \rightarrow XYZ \quad (1)$$

$$\quad | c \quad (2)$$

$$X \rightarrow a \quad (3)$$

$$\quad | Y \quad (4)$$

$$Y \rightarrow b \quad (5)$$

$$\quad | \varepsilon \quad (6)$$

$$\text{NULL} = \{X, Y\}$$

A	FIRST(A)
X	a, b
Y	b
Z	c, a, b

A	FOLLOW(A)
X	a, b, c
Y	a, b, c
Z	

	a	b	c
X	3, 4	4	4
Y	6	5, 6	6
Z	1	1	1, 2

# Exemples

# Exemple ( tableau )

$$S \rightarrow FS \quad (1)$$

$$| Q \quad (2)$$

$$| '( ' S ' ) ' S \quad (3)$$

$$F \rightarrow '!' \quad (4)$$

$$Q \rightarrow '?' \quad (5)$$

« Une session est une séquence de **faits** suivi par une **question** ;  
sous-sessions sont permis »



## Exemple ( tableau )

$$Z \rightarrow S\$ \quad (1)$$

$$S \rightarrow LQ \quad (2)$$

$$| ' ( ' S ' ) ' S \quad (3)$$

$$L \rightarrow FL \quad (4)$$

$$| \varepsilon \quad (5)$$

$$F \rightarrow '!' \quad (6)$$

$$Q \rightarrow '?' \quad (7)$$

Plus ça change, ...

# Implémentation

Grammaire :

$$S \rightarrow F \quad (1)$$

$$| '(' S '+' F ')' \quad (2)$$

$$F \rightarrow 'a' \quad (3)$$

Simple parseur en Python :

`parse.py`

The image features a series of concentric circles in various shades of red, creating a tunnel-like effect that draws the eye toward the center. The circles are set against a dark red background. In the center of the circles is a solid dark blue circle. Overlaid on this central blue circle is the text "That's all Folks!" in a white, elegant cursive script. The text is slightly tilted and positioned so that it appears to be emerging from or resting on the central point of the graphic.

*That's all Folks!*