CTD L&A - TDA3 - 24/11/2020

- exercice 4-2, 4-4 feuille TD n°3
- chapitre 4 : analyse lexicale
  * cours T1-10
  * exercice 1 feuille TD n°4

4.2. Construire un automate qui reconnaît L = (b (aa)* b)*

- Théorème d'Arden
  - L'équation de langages $X = r_1 \cdot X + r_2$
    - a pour solution $X = r_1^* \cdot r_2$ (1)
    - admet une solution unique si $\lambda \notin r_1$ (2)

à voir comme une solution d'une équation de langage de la forme $r_1^* \cdot r_2 = X$ où $r_1 = b(aa)^* b$, $r_2 = \lambda$, $X = L$

$L = (b(aa)^* b)^*$

Arden(1)

$= b(aa)^* b \cdot L_1 + \lambda$ // abstraction

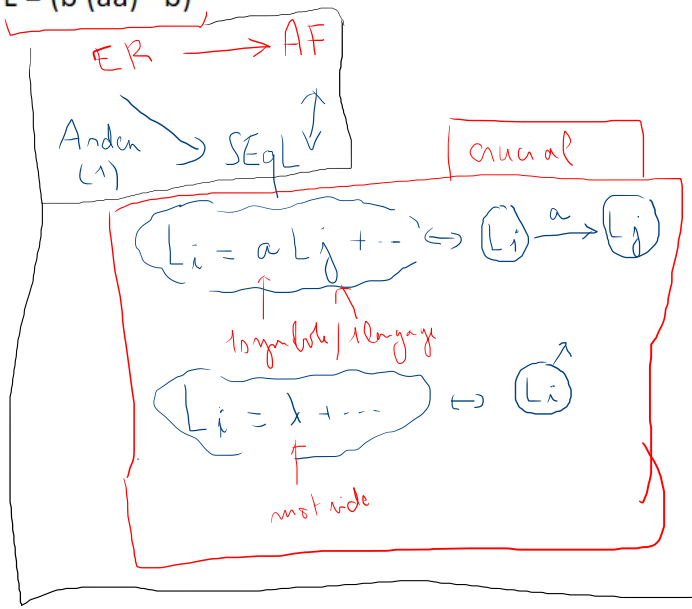
$= b \cdot L_1 + \lambda$ avec $L_1 = (aa)^* b \cdot L$

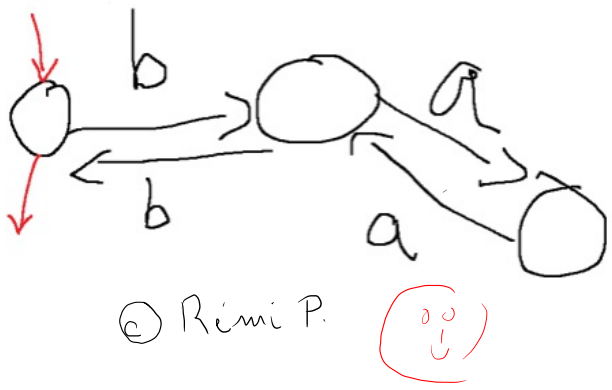
$L_1 = (aa)^* b \cdot L$ de la forme $X = r_1^* \cdot r_2$ où $r_1 = aa$, $r_2 = bL$, $X = L_1$

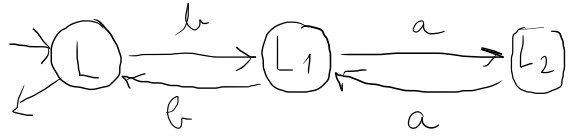
Arden(1)

$= aa L_1 + bL = aL_2 + bL$ avec $L_2 = aL_1$

$L_2 = aL_1$





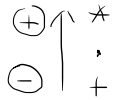


4.4. Construire un automate qui reconnait $J_0 = ba + a^*b^*$

Feuille TD n°3 (mordh)

4.4. Construire un automate qui reconnait $J_0 = ba + a^*b^*$

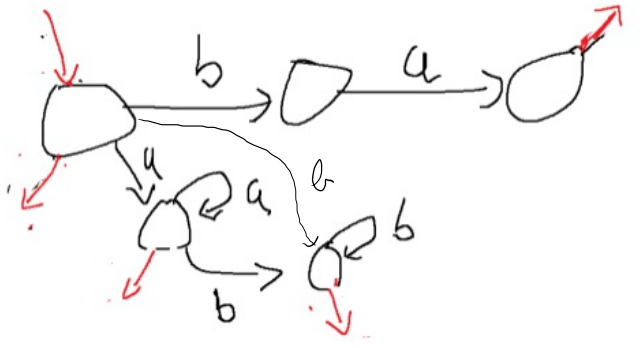
ER → AF



$J_0 = ba$ ou a^*b^*

$= \{ba\} \cup \{w \in X^* \mid w = a^n b^m \text{ où } n, m \in \mathbb{N}\}$

- Théorème d'Arden
  - L'équation de langages $X = r_1 \cdot X + r_2$
    - a pour solution $X = r_1^* \cdot r_2$ (1)
    - admet une solution unique si $\lambda \notin r_1$ (2)



- $J_0 = ba + a^*b^*$
- $= bJ_1 + L_0$ avec $L_0 = a^*b^*$ et avec $J_1 = a$

Arden(1) — 1 mot vide — 1 langage

$L_0 = aL_0 + b^*$

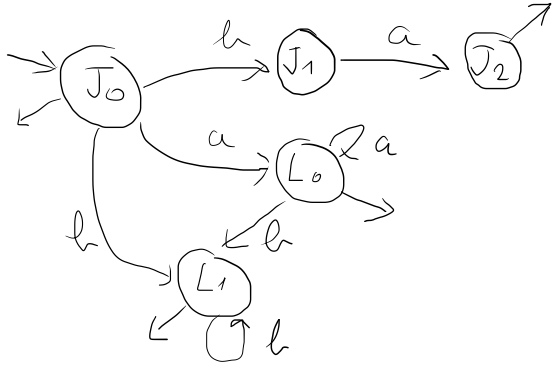
↑ Arden(1) $r_1 = a$, $r_2 = b^*$

$J_0 = bJ_1 + aL_0 + b^*$

$= bJ_1 + aL_0 + L_1$ avec $L_1 = b^*$

$= bL_1 + \lambda$

↑ Arden(1) $r_1 = b$, $r_2 = \lambda$



$J_0 = bJ_1 + aL_0 + bL_1 + \lambda$

- $J_1 = a = aJ_2$ avec $J_2 = \lambda$
- $L_0 = aL_0 + b^* = aL_0 + L_1 = aL_0 + bL_1 + \lambda$

- $L_1 = bL_1 + \lambda$

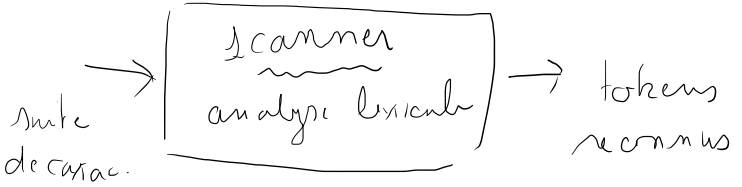
5.6 (chap. 4)

$L_{ENTIER} = (0+1+2+3+4+5+6+7+8+9)\,(0+1+2+3+4+5+6+7+8+9)^*$

ER caractérisant les entiers

AF pour L_{ENTIER}

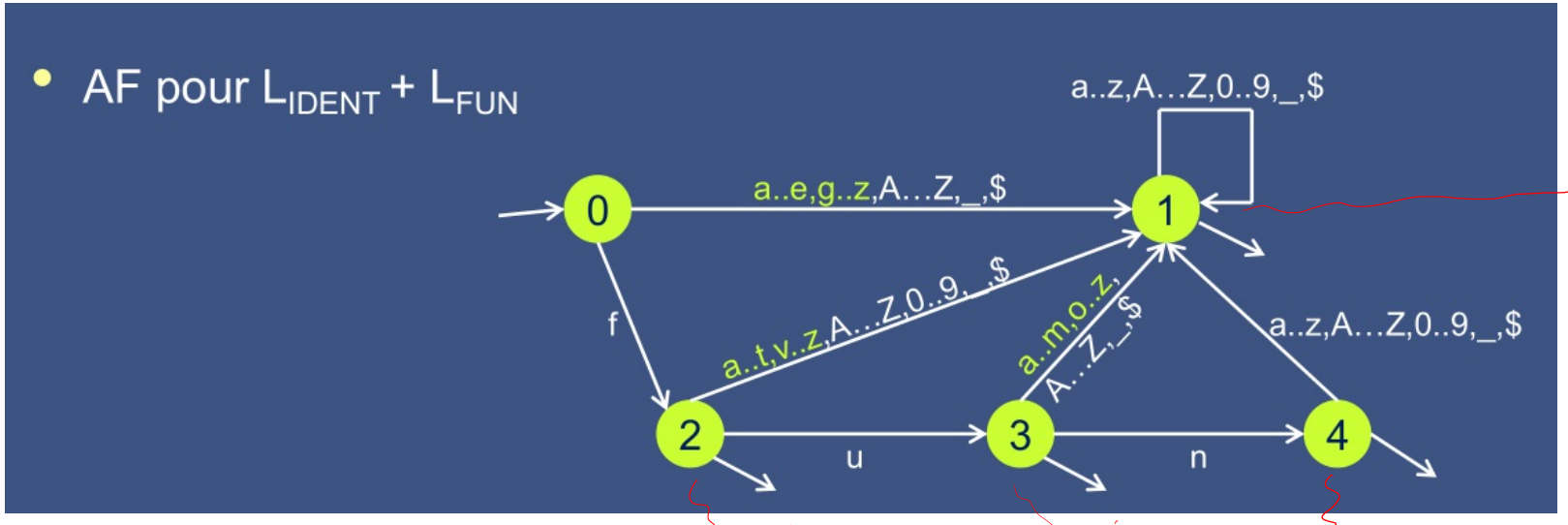
AF correspondant



343 ∈ ENTIER ? OUI 343 ⇝ <ENTIER>

34a ∈ ENTIER ? NON

- AF pour $L_{IDENT} + L_{FUN}$

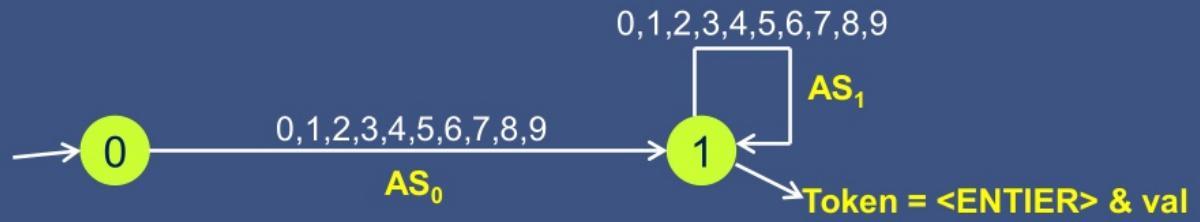


↳ reconnait les suites de caractères qui sont :

- le mot-clé "fun"
- un identificateur.

8

$L_{ENTIER} = (0+1+2+3+4+5+6+7+8+9)(0+1+2+3+4+5+6+7+8+9)^*$



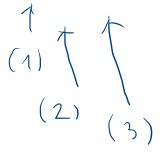
AS_0 : val = asciiToInt(carcour)
AS_1 : val = val * 10 + asciiToInt(carcour)

Remarques

carcour
↳ caractère courant

- AS_0 / AS_1) actions sémantiques !
- les AS sont associées aux transitions de l'AF.

exemple : on lit "343"



(1)  exec AS_0 : val = 3

(2)  exec AS_1 : val = 3 × 10 + 4 = 34

(3)  exec AS_1 : val = 34 × 10 + 3 = 343

fin des caractères ① Token = <ENTIER> et val = 343

exercice 1, feuille TD 4 (moodle)

On considère les constantes numériques entières éventuellement signées suivantes :

(C1)
- Un nombre binaire commence par la chaîne « 0b » ou « 0B » éventuellement précédée par « + » ou « - » et suivie d'une suite non vide de caractères pris dans l'ensemble {0,1}

(C2)
- Un nombre octal commence par le caractère « 0 » éventuellement précédé par « + » ou par « - » et suivi d'une suite non vide de chiffres de l'ensemble {0,1,2,3,4,5,6,7}

(C3)
- Un nombre hexadécimal commence par la chaîne « 0x » ou « 0X » éventuellement précédée par « + » ou « - » et suivie d'une suite non vide de caractères pris dans l'ensemble {0,1,2,3,4,5,6,7,8,9} U {a, b, c, d, e, f} U {A, B, C, D, E, F}.

ou ou non

Construire un analyseur lexical qui fournit le type de la constante avec sa valeur décimale signée et identifie les erreurs syntaxiques.

+/-

of en 1 coup...

1 donner des ER pour les différents types de constantes
2 Construire l'AF correspondant
3 Annoter cet AF avec des AS et des reconnaissances de token
4 gérer les cas d'erreur

1°) Les ER ...

(C1) ⇝ $(\lambda + "+" + "-") . 0 . (b + B) . (0+1) . (0+1)^*$

on démarre par un +, un - ou rien — 0b ou 0B — une suite non vide de 0 et de 1.

(C2) ⇝ $(\lambda + "+" + "-") . 0 . (0+1+2+3+4+5+6+7) . (0+1+2+3+4+5+6+7)^*$

(C3) ⇝ $(\lambda + "+" + "-") . 0 . (x + X) . (0+1+ \dots +9+a+b+\dots+f+A+B+\dots+F) . (0+1+2+\dots+9+a+b+\dots+f+A+B+\dots+F)^*$

2°) écrire 1 AF pr $ER_{C1} + ER_{C2} + ER_{C3}$ réfléchir aux AS

pour la prochaine fois