Théorie des Langages et des Automates (TLA)

- Lobna Kriaa (lobna_kriaa@yahoo.com)
- Bureau B3.

- Examen + CC.

  CC= DS+Tests+Projet

ENSI II1

Objectifs

Ce cours présente la théorie des langages en traitant trois aspects :

1. L'aspect représentation par propriétés mesurables, définitions récursives et expressions régulières.
2. L'aspect reconnaissance par les automates finis, les automates à pile et les machines de Turing.
3. L'aspect génération par les grammaires régulières, non contextuelles et contextuelles,

L'objectif est d'introduire des connaissances en théorie des langages et des automates afin de pouvoir les étendre à la description des langages de programmation et leur analyse syntaxique en vue de leur compilation.

Références.

- A. Aho, R. Sethi et J. Ullman, Compilateurs Principes, Techniques et Outils, InterEditions, Paris, 1991.
- P. Walper, Introduction à la Calculabilité, Dunod, Paris, 2001.
- G. Dowek et J. Lévy, Introduction à la théorie des langages de programmation, Éditions de l'École polytechnique, Paris, 2006.

Langage de Programmation Définir et Reconnaître (1)

1. Définir le langage en terme d'alphabet, mots clés, etc. : Définir les lexèmes
   - Quels sont les mécanismes qui me permettent de reconnaître ces unités lexicales?
2. Définir la syntaxe : Comment écrire l'ensemble des instructions
   - Quels sont les mécanismes qui me permettent de reconnaître la Syntaxe?
3. Définir la sémantique: La relation entre les types
   - Quels sont les mécanismes qui me permettent de définir cette sémantique?
4. Définir quelle machine exécutera le programme décrit dans ce langages de programmation
   - Comment générer du code binaire?

Langage de Programmation Définir et Reconnaître (2)

1. Définir le langage en terme d'alphabet, mots clé, etc. : Définir les lexèmes
   - Quels sont les mécanismes qui me permettent de reconnaitre ces unités lexicales?
2. Définir la syntaxe : Comment écrire l'ensemble des instructions
   - Quels sont les mécanismes qui me permettent de reconnaitre la Syntaxe?
3. Définir la sémantique: La relation entre les types
   - Quels sont les mécanismes qui me permettent de définir cette sémantique?
4. Définir quelle machine exécutera le programme décrit dans ce langages de programmation
   - Comment générer du code binaire?

Alphabet et langages

Automates à états finis

Les Grammaires

Les automates à piles

Langage de Programmation
Définir et Reconnaître (3)

1. Définir le langage en terme d'alphabet, mots clé, etc. : Définir les lexèmes
   - Quels sont les mécanismes qui me permettent de reconnaitre ces unités lexicales?
2. Définir la syntaxe : Comment écrire l'ensemble des instructions
   - Quels sont les mécanismes qui me permettent de reconnaitre la Syntaxe?
3. Définir la sémantique: La relation entre les types
   - Quels sont les mécanismes qui me permettent de définir cette sémantique?
4. Définir quelle machine exécutera le programme décrit dans ce langages de programmation
   - Comment générer du code binaire?

Alphabet et langages

Automates à états finis

Les Grammaires

Les automates à piles

Compilation

Analyse Lexicale

Analyse Syntaxique

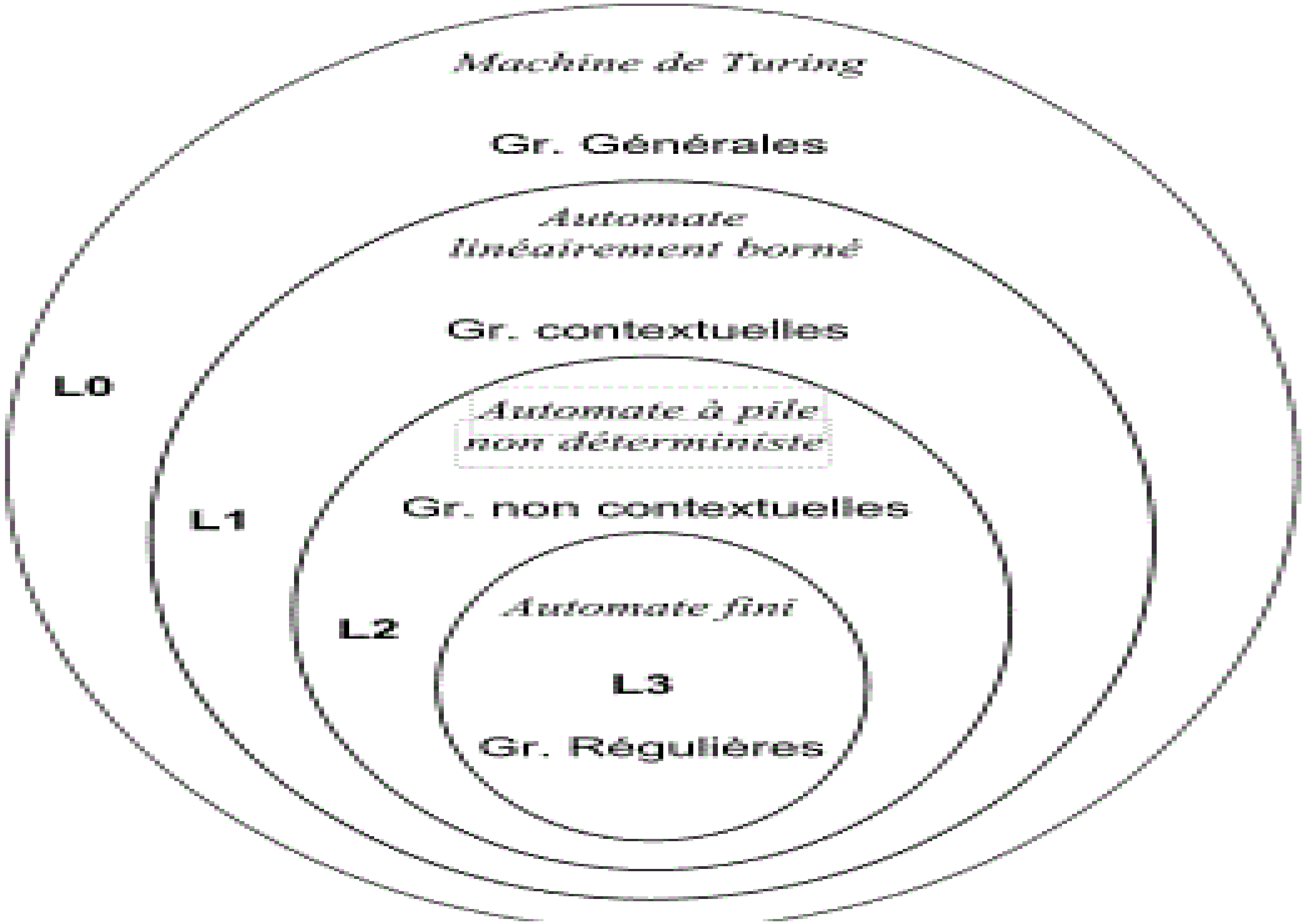
Génération du code Intermediaires

Génération du code cible

Comment reconnaitre les autres langages

Machines de Turing

Classification des langages



Plan Chapitre 1

1. Alphabet et Mot
2. Operations sur les mots
3. Langage
4. Propriétés sur les langages
5. Lemme d'Arden
6. Représentation finie des langages : expressions régulières
7. Loi algébriques sur les expressions régulières
8. Langage régulier
9. Propriétés des langages réguliers

Définitions :

1. ***Lexème ou symbole :*** entité abstraite représentée par un graphique (point, ligne, etc.) exemple : une lettre, un chiffre

2. ***Alphabet :*** Ensemble **fini** non vide de symboles, noté en général par $\sum$

   - Alphabet Latin $\sum=\{a,b,c,\ldots,z\}$
   - Alphabet binaire $\sum=\{0,1\}$
   - $\sum=\{rouge,noir,0,1,a\}$

On note $\sum^*$ l'ensemble de tous mots de $\sum$ (Fermeture de l'alphabet)

3. **Mot ou chaîne** : Séquence de symboles de l'alphabet, noté **w**.

   - w_1= voiture ; w_2 = voyage deux mots définies sur l'alphabet Latin
   - w_1=00101 ; w_2=101101 sont deux mots définies sur l'alphabet binaire.
   - w_1=noir01rouge ; w_2=10aanoir : sont deux mots définies sur l'alphabet $\sum=\{rouge,noir,0,1,a\}$

3. Taille ou longueur d'un mot : soit w €∑*, |w|= nombre de symboles constituant le mot.

- |rouge| = 5 en considérant l'alphabet Latin
- |001|=3 en considérant l'alphabet binaire
- |rouge|=1 en considérant l'alphabet ∑={rouge,noir,0,1,a}

4. Chaîne vide : notée ε s'il n'appartient pas à l'alphabet.

- |ε|=0.
- ε est aussi l'élément neutre de ∑*

5. Sous chaîne : *x* est une sous chaîne de *w* si il existe *y* et *z* (chaînes sur le même alphabet). Tel que *w* = *y x z*.

6. **Préfixe :** x est un préfixe de w si il existe y tel que: $w = x\, y$.
7. **Suffixe :** x est un suffixe de w si il existe y tel que $w = y\, x$.
    - *Exemple :* w=001 ;
        - x=00 est prefixe car il existe y =1/ w=xy=001
        - X=1 est un suffixe de w car il existe y =00 / w=yx=001
8. ***Facteur :*** soit u,v,w,t des mots définis sur $\sum$ tel que $w = uvt$
    - si $u = \varepsilon$ alors v est dit facteur gauche de w (ou préfixe).
    - si $t = \varepsilon$ alors v est dit facteur droit de w (ou suffixe).
    - si $u = t = \varepsilon$ alors w est un facteur de lui même.

Concaténation : soient u et v deux mots définis sur l'alphabet ∑, tel que :

$$u = x_1 x_2 \ldots x_n \qquad v = y_1 y_2 \ldots y_m$$

$$w = u.v = x_1 x_2 \ldots x_n y_1 y_2 \ldots y_m$$

La concaténation *est non commutative*

Propriétés :

- *W=(uv)t=u(vt), la concaténation est associative*
- *Si w= uv alors* $|w|=|uv|=|u|+|v|$
    - *Ex:* $|w^n|=n.\,|w|$ $\qquad w^n = \underbrace{ww\ldots w}_{n \text{ fois}}$
- *ε est l'élément neutre pour la concaténation.* $\varepsilon x = x\varepsilon = x$

9. Occurrence d'un symbole dans un mot :

Le nombre d'occurrences d'un symbole x dans un mot ω est le nombre de fois où ce symbole apparaît dans ce mot ω. On le note $|\omega|_x$.

$$|\omega| = \sum_{x\in\Sigma} |\omega|_x$$

Exercice :

Quel est le nombre d'occurrences de b dans les mots abba et ε

Corrigé :

$|abba|_b=2$

$|\varepsilon|_b=0$

$|abaaba|_a=4$

10. Image (reverse) : *w=aabab* $w^R=babaa$

- Si $x \in \Sigma$ alors $x^R=x$
- Si $w=xu$ alors $w^R=u^Rx$

Monter que $(wu)^R = u^R w^R \quad \forall k / |u| = k$

Par induction sur la longueur de u .

- $|u|=1$, $(wu)^R=uw^R=u^Rw^R$.
- Supposons que c'est vrai pour $k>1$ et montrons le pour $n>k$.
- $u=tx$, avec $|u| \geq k$ et $|t| <k$, $x \in \Sigma$.
- $(wu)^R= (wtx)^R=x^R(wt)^R=xt^Rw^R$, par HI, $|t| \leq k$.

$$=x^Rt^Rw^R= (tx)^Rw^R= u^Rw^R$$

• Σ^k = ensemble des mots de longueur k avec des symboles de Σ

Exemple :

• $\Sigma = \{0,1\}$, $\Sigma^0 = \{\varepsilon\}$, $\Sigma^1 = \{0,1\}$, $\Sigma^2 = \{00,01,10,11\}$

• Σ^* : tous les mots sur Σ y compris ε

$$\Sigma^* = \{\varepsilon\} \cup \Sigma^1 \cup \Sigma^2 \ldots$$

• Σ^* est infinie et dénombrable

$$\Sigma^+ = \Sigma^1 \cup \Sigma^2 \ldots$$

• On note Σ+ tous les mots sur Σ sans ε

$$\Sigma^+ \cup \{\varepsilon\} = \Sigma^*$$

• Si $\alpha\beta = \varepsilon$ alors $\alpha = \beta = \varepsilon$

• Une factorisation d'un mot est son écriture sous forme d'un produit explicite de facteurs

Définition : Langage

- On appelle langage tout ensemble de mots
- On définit un langage sur un alphabet Σ comme un sous ensemble de Σ *

Exemple:

- Si le vocabulaire est $\Sigma = \{0, 1, 2, 3, \ldots 9\}$

  L = { représentations décimales des nombres entiers naturels}

  L = |N
- Si le vocabulaire est $\Sigma = \{x1, x2, +, *, (,)\}$

  L = {expressions arithmétiques parenthèsées manipulant x1, x2, + et * }

Si Σ est un alphabet, et $L \subseteq \Sigma^*$ alors L est un langage

Exemples de Langages

- Si $\sum$ est la vocabulaire du langage de programmation C

  $\sum$ = {**main, (,), Include, #, <, >, ., ;, id, nb, …….**}

  L= {**programmes C corrects syntaxiquement**}

- Si $\sum$ est le vocabulaire de la logique des propositions
- $\sum$ ={**p, (,), $\rightarrow$, $\wedge$, $\neg$**} où p désigne une proposition

  L = {**formules bien formées de la logique des propositions**}

•Ensemble des mots de l'alphabet binaire contenant un nombre de *n* de 0 suivie par le même nombre *n* de 1.

L={ε ; 01 ; 0011; 000111; … }.

- Ensemble des mots de l'alphabet binaire ayant un même nombre de 0 et de 1.

L={ε ; 01 ; 10; 0101; 1001; … }.

- Ensemble des mots de l'alphabet binaire tel que leur valeur est premier.

L={10; 11; 101; 111; 1011; …}

Définition des langages

Définition par propriété mesurable

Un langage c'est un ensemble de mots appartenant à $\sum^*$ et qui vérifie une propriété donnée : L= {w $\in \sum^*$ | w possède la propriété P}

- L est l'ensemble des mots sur {a, b} de longueur paire

  $L = \{ \omega \in \{a, b\}^* / |\omega| = 2k, K \geq 0\}$

- L est l'ensemble des mots sur {a, b} ayant un nombre impaire de b

  $L = \{ \omega \in \{a, b\}^* / |\omega|b = 2k+1, K \geq 0\}$

- L est l'ensemble des mots sur {a, b} où tous les a précèdent les b et sont de même nombre

  $L = \{\{ \omega \in \{a, b\}^* / a^n b^n, n \geq 0\}$

Définition des langages

Définition récursive : Définition dans laquelle, un langage est définie sur lui même.

Exemples

- $L_2=\{w \in \sum *| \ w = a \text{ ou } w = aw_1;\ w_1 \in L_2\}=\{a,aa,\ldots,aaaa,\ldots\}$
- $L_3=\{w \in \sum *| \ w = \varepsilon \text{ ou } w = w_1w_2;\ |w_1| =2 \text{ et } w_2 \in L_3\}$

- L est l'ensemble des mots sur {a, b} où tous les a précèdent les b et sont de même nombre
  - ✓La définition par propriété mesurable est la suivante :

    $$\mathbf{L = \{ a^n b^n, n \geq 0\}}$$

  - ✓La définition récursive du même langage est :

    $$\mathbf{L = \{ \omega \in \{a, b\}^* / \omega = \varepsilon \text{ ou } \omega = a\omega_1 b \text{ et } \omega_1 \in L\}}$$

- Le langage vide L= $\varnothing$;
- Le langage $\{\varepsilon\}$ contenant le mot vide.

Note: $\varnothing \neq \{\varepsilon\}$.

Note: L'alphabet Σ est un ensemble fini.

- Ensemble des palindromes sur l'alphabet $\Sigma = \{a,b\}$

$$L = \{w \in \Sigma^* \mid w = w^R\}$$

$$L = \{\varepsilon, aba, bab, a, b, \ldots\}$$

Exercice

Definir le langage d'une manière récursive des palindromes sur l'alphabet {a,b}

a- De longueur paire

b- De longueur impaire

Propriétés sur les langages

- $L_1 \backslash L_2$: = $\{w \in \Sigma^* \mid w \in L_1$ ou w n'appartient pas à $L_2 \}$
- $L = L_1 \cup L_2 = \{w \in \Sigma^* \mid w \in L_1$ ou $w \in L_2 \}$
  - **L ∪ M = M ∪ L. Union est commutative.**
  - **(L ∪ M) ∪ N = L ∪ (M ∪ N). Union est associative.**
- $L = L_1 \cap L_2 = \{w \in \Sigma^* \mid w \in L_1$ et $w \in L_2 \}$
- $\overline{L}$: complèment $\Sigma^* \backslash L$
- Concaténation :

$$L = L_1 \,.\, L_2 = L_1L_2 = \{w \in \Sigma^* \mid \exists\, x , y , w = x\, y , x \in L_1, y \in L_2 \}$$

  - **(LM)N = L(MN) :Concaténation est associative**
  - **Note: Concaténation n'est pas commutative. Il existe L et M tel que LM ≠ ML.**

Propriétés sur les langages

- Fermeture de Kleene.

  $L^* = \{w \in \Sigma^* | \ w = w_1 w_2 \ldots w_k,\ k \geq 0$ et $w_1, w_2, \ldots, w_k \in L\}$

  $k = 0 \Rightarrow w = \varepsilon$; $k = 1 \Rightarrow w \in L$;

  Si L est un langage (ensemble de mots) alors L* désigne l'ensemble de toutes les chaînes de longueurs finies formées par concaténation de mots de L, où chaque mot peut être utilisé de 0 à n fois, la chaîne vide est incluse.

-

Propriétés sur les langages

- $L(M \cup N) = LM \cup LN$.

  Concaténation est distributive à gauche pour l'union.

- $(M \cup N)L = ML \cup NL$.

  Concaténation est distributive à droite pour l'union.

- $L \cup L = L$.

  Union est idempotent.

- $\varnothing^* = \{\varepsilon\}$, $\{\varepsilon\}^*=\{\varepsilon\}$
- $L^+ = LL^* = L^*L$, $L^*= L^+ \cup\{\varepsilon\}$
- $(L^*)^* = L^*$. Fermeture est idempotente

Propriétés sur les langages

Exemple :

- L={aa,b}
- L*={ε,b,aa,bb,aab,baa,bbb,aaaa,aabb,baab,bbaa,bbbb,aaaab,aabaa,aabbb,baaaa,bbaab,bbbaa,bbbbb,…}

Note : $\emptyset^* = \{\varepsilon\} \neq \emptyset$

Exercice

Soit $\Sigma = \{0,1\}$

L={*w* ∈ Σ*| *w contient un nombre de 1 différent de nombre de 0.*}

Montrer que $L^* = \Sigma^*$

Il y a une différence?

Il faut faire la différence entre:

ε – la chaîne vide ("")

$\varnothing$ – l'ensemble vide({ })

$\{\varepsilon\}$ – l'ensemble qui contient tout simplement la chaîne vide.

Exercice

Calculer A* pour chacun des ensembles A suivants:

1) $A = \{a\}$
2) $A = \{\omega \in X^* \,/\, |\omega| = 2k+1 \,/\, k \geq 0\}$

Corrigé :

1) Si $A = \{a\}$ alors $A^* = \{a\}^* = a^*$

Car $A^* = A^0 + A^1 + A^2 + \ldots\ A^i + \ldots$

$A^0 = \{\varepsilon\} = \{a^0\}$

$A^1 = AA^0 = \{a\}\ \{\varepsilon\} = \{a\} = \{a^1\}$

$A^2 = AA^1 = \{a\}\{a\} = \{aa\} = \{a^2\}$

$A^i = \{a^i\}$

$A^{i+1} = A\ A^i = \{a\}\{a^i\} = \{a^{i+1}\}$

….

$A^* = \{\varepsilon, a, aa, aaa, \ldots\} = \{a^0, a^1, a^2, a^3, \ldots\} = a^*$

Exercice (suite)

2) Si $A = \{\omega \in X^* \ / \ |\omega| = 2k+1 \ / \ k \geq 0\}$

$A^* = X^*$

Car $A^* = A^0 + A^1 + A^2 + \ldots A^i + \ldots$

$A^0 = \{\varepsilon\}$

$A^1 = AA^0 = A\{\varepsilon\} = \{\omega \in X^* \ / \ \mathbf{|\omega| = 2k+1 \ / \ k \geq 0}\}$

$A^2 = AA^1 = \{\omega \in X^* \ / \ |\omega| = 2k+1/ \ k \geq 0\} \ \{\omega \in X^* \ / \ |\omega| = 2k+1/ \ k \geq 0\}$

$= \{\omega \in X^* \ / \ |\omega| = 2k+2/ \ k \geq 0\}$

$= \{\omega \in X^* \ / \ |\omega| = 2k \ / \ k > 0\}$

$A^0 + A^2 = \{\varepsilon\} + \{\omega \in X^* \ / \ |\omega| = 2k \ / \ k > 0\} = \{\omega \in X^* \ / \ \mathbf{|\omega| = 2k/ \ k \geq 0}\}$

$A^0 + A^1 + A^2 = X^*$

Donc $A^* = X^*$

Exercices

Exercice :

Calculer $A.\phi$ et $A.\{\varepsilon\}$

Montrer qu'on n'a pas $A.(B\cap C) = A.B \cap A.C$

Corrigé:

1- $A.\phi = \phi.A = \phi$

2- $A.\{\varepsilon\} = \{\varepsilon\}.A = A$

5- On <u>n'a pas</u> $A.(B\cap C) = A.B \cap A.C$

Contre exemple : $A = \{\varepsilon, x\}$, $B = \{xyzt\}$, $C = \{yzt\}$

Exercice

Contre exemple : $A = \{\varepsilon, x\}$, $B = \{xy\}$, $C = \{y\}$

Car $A.(B \cap C) \neq A.B \cap A.C$

En effet $(B \cap C) = \phi$ donc $A.(B \cap C) = \phi$

Par contre $A.B = \{xy, xxy\}$ et $A.C = \{y, xy\}$ donc $A.B \cap A.C = \{xy\}$

6- On n'a pas **$A^+ = A^* - \{\varepsilon\}$** par contre on $A^* = A^+ + \{\varepsilon\}$

Pour $A = \{\varepsilon, a\}$

$A^+ = A^1 + A^2 + \ldots = \{\boldsymbol{\varepsilon}, a\} + \{\varepsilon, a, aa\} + \ldots$

$A^* = A^0 + A^1 + A^2 + \ldots = \{a^i \mid i \geq 0\}$

$A^* - \{\varepsilon\} = \{a^i \mid i > 0\} \neq A^+$ car $A \supseteq \{\varepsilon\}$

$A^+ = A^* - \{\varepsilon\}$ est vraie lorsque A ne contient pas ε

Exercise

Completer

$L^+ \bullet \{\varepsilon\}$ = ___________

$\{\varepsilon\} \bullet \{\varepsilon\}$ = ___________

$\varnothing \bullet L$ = ___________

$L^* \bullet L^*$ = ___________

$(L^*)^*$ = ___________

$L \bullet L^*$ = ___________

$\varnothing^*$ = ___________

$\{\varepsilon\}^*$ = ___________

Exercise

Completer

$L^+ \bullet \{\varepsilon\}$ = ____L+______

$\{\varepsilon\} \bullet \{\varepsilon\}$ = ___ {ε} _______

$\varnothing \bullet L$ = __ ∅ ________

$L^* \bullet L^*$ = ___L*_______

$(L^*)^*$ = ______L*_____

$L \bullet L^*$ = ___L+_______

$\varnothing^*$ = __{ε}________

$\{\varepsilon\}^*$ = ______{ε}_______

Lemme d'Arden

Pour deux langages A et B d'un vocabulaire $\sum^*$,

Les équations **$L= AL+B$** et **$L = LA+B$** admettent respectivement comme solution minimale A^*B et BA^*. Cette solution est unique si $\varepsilon \notin A$.

Expressions régulières :

Notation:

$a^k =$ aaaaaaaaa…… $a^2 = aa$

<---------------->

k fois

$a^0 = \varepsilon$

$a^* = \{a^i \mid i \geq 0\} = \{a^0, a^1, a^2, \ldots.a^i, \ldots\}$

$a^+ = \{a^i \mid i > 0\} = \{a^1, a^2, \ldots., a^i, \ldots\}$

$L_4=\{\varepsilon, x, xx, xxx, xxxx, \ldots.\}$

soit S = {*x*} alors $L_4=S^*$ ou $L_4=\{x\}^*$

Considérons l'étoile de la fermeture de Kleene appliquée à la lettre *x*. *x**

- *x** indiquera une séquence quelconque de x qui peut être vide.
- *x** = ε ou *x* ou *xx* ou *xxx*… L_4 = langage (x*)

- Considérons le langage

L = {*a*,*ab*,*abb*,*abbb*,*abbbb*,…}

Toutes les chaînes constitués par un *a* suivi d'un nombre quelconque de *b*

On peut noter : L=Langage(*ab**)

Langage dans lequel les mots sont la concaténation d'un *a* (*a*) initial avec un nombre quelconque de *b* (*b**).

Appliquons l'étoile de Kleene à toute la chaîne *ab,* on aura :

(*ab*)*= ε ou *ab* ou *abab ou ….*

- Le langage définit par l'expression :

$$ab^*a$$

Ensemble de toutes les chaînes de *a* et de *b* qui ont au moins deux lettres, qui commencent et finissent par un *a*. et qui n'ont que des *b* ou rien à l'intérieur.

langage (ab^*a)={aa,aba,$abba$,$abbba$,…}

Remarque :

Fausse description : Ensemble de tous les mots qui commencent et puis finissent par *a* et qui n'ont que des *b* (ou rien) entre eux.

Le mot *a* appartient a cette description.

- Le langage définit par l’expression :

$$a^*b^*$$

Ensemble de toutes les chaînes de *a* et de *b dans lesquelles les a’s viennent avant les b’s.*

langage (a^*b^*)=$\{\varepsilon, a, b, aa, bb, ab, bb, aaa, abb, \ldots\}$

Remarque :

$a^*b^* \neq (ab)^*$

Le langage à droite contient *abab* tandis que celui à gauche ne le contient pas

T définie sur $\Sigma=\{a,b,c\}$

$T=\{a,c,ab,cb,abb,cbb,abbb,cbbb,abbb,cbbbb,\ldots\}$

• Tous les mots de T commencent avec un *a* ou un *c* ensuite ils sont suivis par un nombre quelconque éventuellement nulle de b.

Symboliquement T=langage $((a\cup c)b^*)$

Exemple :

• Ensemble des chaînes de *a* et de *b* de longueur 3.

• Ensemble des chaînes de *a* et de *b* de longueur quelconque

Expression régulière
Définition

Une ***expression régulière*** sur un alphabet Σ est une chaîne de caractère sur l'alphabet Σ,(,), $\cup$,*,$\emptyset$. Tel que :

1. Toute lettre de $\Sigma \cup \{\varepsilon\}$ et $\emptyset$ est une expression régulière

2. si r_1 et r_2 sont deux expressions régulières alors ;

- (r_1)
- r_1r_2
- $r_1 \cup r_2$
- r_1^*

Sont des expressions régulières

3. Rien d'autre n'est une expression régulière.

Exemples d'ER

- a^*b^*
- $(a+b)^*c^+$
- $(0+1+2+\ldots 9)^+$ **représentations décimales des entiers**
- $(a+b)^*aa$ **mots sur {a, b} ayant aa comme facteur droit**
- $(a+b+c)^*abc(a+b+c)^*$ **mots sur {a,b,c} ayant abc comme facteur**
- $0^*(1+2+\ldots 9)(0+1+\ldots 9)^*$ **représentations décimales des entiers non nuls**
- $(0+1+2+\ldots 9)^*1(0+1+2+\ldots 9)$
  représentation décimale des entiers naturels ayant 1 dans les dizaines
- $(0+1+2+\ldots 9)^*10$ **représentation décimale des entiers naturels ayant 1 dans les dizaines et 0 dans les unités**
- $(a+b+\ldots z+A+B+\ldots Z)(a+b+\ldots z+A+B+\ldots Z+0+1+\ldots 9)^*$
  les identificateurs alphanumériques qui commencent par un caractère alphabétique

Lois algébriques sur les ER

Soit r1, r2, r3 des expressions régulières

- $r_1 . \varepsilon = \varepsilon . r_1 = r_1$
- $r_1^* \cup \varepsilon = r_1^*$
- $r_1 \cup r_2 = r_2 \cup r_1$
- $(r_1 \cup r_2) \cup r_3 = r_1 \cup (r_2 \cup r_3)$
- $(r_1 . r_2) . r_3 = r_1 . (r_2 . r_3)$
- $(r_1 \cup r_2) . r_3 = (r_1 . r_3) \cup (r_2 . r_3)$
- $r_1 r_1^* = r_1^+$
- $r_1 \cup r_1 = r_1$
- $\varnothing r_1 = r_1 \varnothing = \varnothing$
- $\varnothing \cup r_1 = r_1 \cup \varnothing = r_1$

Expression régulière exemples

Exercice 1 :

$\Sigma=\{a,b\}$

$L=\{w \in \Sigma^* \mid w$ contient la sous chaîne $aa\}$

Exercice 2 :

$\Sigma=\{a,b\}$

$L=\{w \in \Sigma^* \mid w$ *ne* contient pas 3 b consécutifs $\}$

Théorème :

Un langage L est dit régulier si et seulement si il existe une expression régulière qui le génère.

Exemple :

$\Sigma=\{a,b\}$

L=$\{w \in \Sigma^*|\ w$ contient un nombre paire de a et un nombre pair de $b\ \}$

Propriétés

Étant donné deux langages réguliers L_1 et L_2

- $L_1 \cup L_2$: est un langage régulier
- $L_1 . L_2$: est un langage régulier
- L_1***** : est un langage régulier
- $\overline{L_1} = \Sigma^* \setminus L_1$ est un langage régulier
- $L_1 \cap L_2 = \overline{(\overline{L_1} \cup \overline{L_2})}$ est un langage régulier
- L^R : est un langage régulier
- $L_1 \setminus L_2$: est un langage régulier

Définition : Deux expressions régulières α et β sont dites équivalentes si $L(\alpha) = L(\beta)$

Autrement s'ils génèrent le même langage

Exemple :

Langage de tous les mots qui ont au moins 2 a's peut être décrit par l'expression régulière :

$$(a \cup b)^*a\,(a \cup b)^*a\,(a \cup b)^*$$

Autre expression régulière

$$b^*ab^*a(a \cup b)^*$$

On peut noter :

$$(a \cup b)^*a\,(a \cup b)^*a\,(a \cup b)^* = b^*ab^*a(a \cup b)^*$$