# Automates, langages et compilation

## Langages et expressions rationnelles

Isabelle Ryl

2024 – 2025

Cours de L3 - Université Paris Dauphine-PSL

1. Langages

2. Langages et expressions rationnelles

# Langages

# Alphabet

Un **alphabet** est un ensemble de symboles

Exemples

- $\Sigma_0 = \{a, b, c\}$ ou $\Sigma_1 = \{i, f, a, g, c, o\}$
- mais aussi $\Sigma_2 = \{0, a, *, (\}$
- ou $\Sigma_3 = \{\mho, \bowtie, \square, \diamond, \rightsquigarrow\}$
- et même $\Sigma_4 = \{\mathit{bonjour}, \mathit{coucou}, \mathit{hello}\}$
- ou plutôt $\Sigma_5 = \{\mathit{if}, \mathit{or}, \mathit{print}, =, \mathit{true}, v, \mathit{false}, \mathit{then}\}$

# Mot

Un **mot** est une suite de symboles. Un mot est défini sur un alphabet.
Exemples :

- sur l'alphabet $\Sigma_0$ : $a, aa, aaa, abc, abbbbaaaaacccc$
- sur l'alphabet $\Sigma_3$ : $\mho, \bowtie \mho, \mho \square \mho \bowtie\bowtie \diamond \rightsquigarrow$
- sur l'alphabet $\Sigma_5$ : $if\ v = true$

On note le **mot vide** : $\varepsilon$

La **longeur d'un mot** est le nombre de symboles qui le composent :

- si $abc$ est un mot sur l'alphabet $\Sigma_0$, $|abc| = 3$
- si $if$ est un mot sur l'alphabet $\Sigma_1$, $|if| = 2$
- si $if\ v = true$ est un mot sur l'alphabet $\Sigma_5$, $|if\ v = true| = 4$
- $|\varepsilon| = 0$

# Parenthèse

- Durant l'analyse lexicale, on identifie les éléments du langage de programmation, les mots clés, les entiers, les noms de variables, etc. L'alphabet est l'ensemble des symboles du clavier qui peuvent être utilisés, les mots sont par exemple « if », « 538 », « + », etc
- Durant l'analyse syntaxique, l'alphabet est composé des mots clés, entiers, noms de variables, etc (*i.e.* les mots de l'analyse lexicale) et les mots sont des suites de ceux-ci et donc des programmes

# Mot

- Soit $\Sigma$ un alphabet, on note $\Sigma^*$ l'ensemble des mots que l'on peut construire sur l'alphabet $\Sigma$, y compris le mot vide
- Soit $\Sigma$ un alphabet, on note $\Sigma^+$ l'ensemble des mots non vides que l'on peut construire sur l'alphabet $\Sigma$
- Soit $u \in \Sigma^*$ tel que $u = x_0 x_1 \dots x_n$ avec $\forall 0 \leq i \leq n, x_i \in \Sigma$,
  - $x_n x_{n-1} \dots x_0$ est appelé **inverse** de $u$, noté $u^R$, on peut le définir de manière inductive :
    - si $|u| = 0$, alors $u^R = \varepsilon = u$
    - si $|u| > 0$, alors $u = vx_n$ avec $v \in \Sigma^*$ et $u^R = x_n v^R$
  - si $u$ est égal à son inverse alors $u$ est un **palindrome** *i.e.* un mot qui peut se lire dans les deux sens, par exemple RADAR

# Concaténation

Soit $u = x_0 x_1 \dots x_n$ et $v = y_0 y_1 \dots y_m$ deux mots définis sur l'alphabet $\Sigma$ où les $x_0 \dots x_n$ et les $y_0 \dots y_m$ sont des lettres de $\Sigma$. La **concaténation** de $u$ et $vv$, notée $u.v$ ou $uv$, est le mot formé de la suite des lettres de $u$ suivie de la suite des lettres de $v$ soit : $uv = x_0 x_1 \dots x_n y_0 y_1 \dots y_m$.

- Associativité : $\forall u, v, w \in \Sigma^*, (uv)w = u(vw)$
- Élément neutre : $\forall u \in \Sigma^*, u\varepsilon = \varepsilon u = u$
- Puissance :
  - $\forall u \in \Sigma^*, u^0 = \varepsilon$
  - $\forall u \in \Sigma^*, \forall n \in \mathbb{N}, n > 0, u^n = uu^{n-1}$

# Quelques définitions de mots

Soient $u, v \in \Sigma^*$ :

- $u$ est un **facteur** de $v$, si $\exists w_1, w_2 \in \Sigma^*$, tels que $v = w_1 u w_2$
- $u$ est un **préfixe** de $v$, si $\exists w \in \Sigma^*$, tel que $v = uw$
- $u$ est un **suffixe** de $v$, si $\exists w \in \Sigma^*$, tel que $v = wu$
- $u$ est un **sous-mot** de $v$, si $\exists u_0, \ldots, u_n, v_0, \ldots, v_{n+1} \in \Sigma^*$ tels que $u = u_0 u_1 \ldots u_n$ et $v = v_0 u_0 v_1 u_1 \ldots v_n u_n v_{n+1}$

Dans chacun des cas, si $u \neq v$, $u$ est dit facteur/préfixe/suffixe/sous-mot **propre** de $v$

# Exercice

Question Montrer que la relation « être préfixe de » est une relation d'ordre partiel sur $\Sigma^*$

- Réflexivité : $\forall u \in \Sigma^*, u = u\varepsilon$ avec $\varepsilon \in \Sigma^*$ donc $u$ est un préfixe de $u$
- Antisymétrie : soient $u, v \in \Sigma^*$ tels que $u$ est préfixe de $v$ et $v$ est préfixe de $u$. Par définition, $\exists u', v' \in \Sigma^*$ tels que $u = vv'$ et $v = uu'$. Donc $u = vv' = uu'v'$ , ce qui implique que $u' = v' = \varepsilon$ et donc $u = v$
- Transitivité : soient $u, v, w \in \Sigma^*$ tels que $u$ est préfixe de $v$ et $v$ est préfixe de $w$, montrons que $u$ est préfixe de $w$. Par définition, $\exists u', v' \in \Sigma^*$ tels que $v = uu'$ et $w = vv'$, donc $w = vv' = uu'v'$ donc $u$ est préfixe de $w$.

Exercices : montrer que les relations « être suffixe de » et « être facteur de » sont des relations d'ordre partiel sur $\Sigma^*$

# Langage

Un **langage** sur un alphabet $\Sigma$ est un ensemble de mots définis sur $\Sigma$, soit un sous-ensemble de $\Sigma^*$

On note :

- $\emptyset$ le langage vide (*i.e.* qui ne contient aucun mot)
- $\{\varepsilon\}$ le langage qui contient uniquement le mot vide

# Exemples

Soit $\Sigma = \{a, b, c, d\}$, quelques exemples de langages sur $\Sigma$ :

- $\mathcal{L}_0 = \{aa, abbdbbb, caba\}$
- $\mathcal{L}_1 = \{a, aa, aaa, aaaa, aaaaa\}$
- $\mathcal{L}_2 = \{u \in \Sigma^* \mid u = \varepsilon \vee u = av \text{ avec } v \in \mathcal{L}_2\}$

# Opérations sur les langages

Soit $\mathcal{L}_1$ et $\mathcal{L}_2$ deux langages définis respectivement sur $\Sigma_1$ et $\Sigma_2$

L'**union** de $\mathcal{L}_1$ et $\mathcal{L}_2$ est $\mathcal{L}_1 \cup \mathcal{L}_2 = \{u \mid u \in \mathcal{L}_1 \vee u \in \mathcal{L}_2\}$

L'**intersection** de $\mathcal{L}_1$ et $\mathcal{L}_2$ est $\mathcal{L}_1 \cap \mathcal{L}_2 = \{u \mid u \in \mathcal{L}_1 \wedge u \in \mathcal{L}_2\}$

Le **complémentaire** de $\mathcal{L}_1$ dans $\Sigma_1$ est $C(\mathcal{L}_1) = \{u \in \Sigma_1^* \mid u \notin \mathcal{L}_1\}$

La **différence** de $\mathcal{L}_1$ et $\mathcal{L}_2$ est $\mathcal{L}_1 - \mathcal{L}_2 = \{u \in \mathcal{L}_1 \mid u \notin \mathcal{L}_2\}$, le langage est donc défini sur $\Sigma_1$

☛Il s'agit d'opérations ensemblistes

# Concaténation de deux langages

La **concaténation** ou le **produit** de deux langages $\mathcal{L}_1$ et $\mathcal{L}_2$ respectivement définis sur $\Sigma_1$ et $\Sigma_2$ est le langage défini sur $\Sigma_1 \cup \Sigma_2$

$$\mathcal{L}_1.\mathcal{L}_2 = \{uv \mid u \in \mathcal{L}_1 \wedge v \in \mathcal{L}_2\}$$

## Exemples

Soit $\mathcal{L}_1 = \{ab, c, a\}$, $\mathcal{L}_2 = \{bb, cc\}$ et $\mathcal{L}_3 = \{c, bc\}$,

$$\mathcal{L}_1.\mathcal{L}_2 = \{abbb, cbb, abb, abcc, ccc, acc\}$$

$$\mathcal{L}_1.\mathcal{L}_3 = \{abc, cc, ac, abbc, cbc\}$$

## Propriétés

- associative : $(\mathcal{L}_1.\mathcal{L}_2).\mathcal{L}_3 = \mathcal{L}_1.(\mathcal{L}_2.\mathcal{L}_3)$
- non commutative, exemple : si $\mathcal{L}_1 = \{a\}$ et $\mathcal{L}_2 = \{b\}$ alors $\mathcal{L}_1.\mathcal{L}_2 = \{ab\}$ et $\mathcal{L}_2.\mathcal{L}_1 = \{ba\}$

# Puissance d'un langage

Les **puissances** d'un langages sont les concaténations successives du langage avec lui-même, elles sont définies récursivement

- $\mathcal{L}^0 = \{\varepsilon\}$
- $\forall n > 0, \mathcal{L}^n = \mathcal{L}.\mathcal{L}^{n-1}$

Exemple

Soit $\mathcal{L} = \{a, b\}$, $\mathcal{L}^3 = \{aaa, baa, aba, bba, aab, bab, abb, bbb\}$

# Étoile d'un langage

L'**étoile** de Kleene ou fermeture de Kleene d'un langage $\mathcal{L}$ est

$$\mathcal{L}^* = \bigcup_{i \geq 0} \mathcal{L}^i$$

Remarques

- on définit également $\mathcal{L}^+ = \bigcup_{i \geq 1} \mathcal{L}^i$
- pour tout langage $\mathcal{L}$, $\varepsilon \in \mathcal{L}^*$
- pour tout langage $\mathcal{L}$, $(\varepsilon \in \mathcal{L}^+) \Leftrightarrow (\varepsilon \in \mathcal{L})$

# Exemples de langages

Soient $\mathcal{L}_0 = \{a\}$, $\mathcal{L}_1 = \{ab, cb\}$, $\mathcal{L}_2 = \{cb\}$

Calculer :

- $\mathcal{L}_0^* = \{\varepsilon, a, aa, aaa, aaaa, aaaaa, \dots\}$

  ➥ les mots composés de $a$

- $\mathcal{L}_1^+ = \{ab, cb, abab, cbcb, abcb, cbab, \dots\}$

  ➥ les mots de longueur paire et non nulle composés de $a$, $b$, $c$, dont les lettres aux places paires sont des $b$

- $\mathcal{L}_1^+ \cap \mathcal{L}_2^* = \mathcal{L}_2^+$

# Démonstration « simple » par récurrence : palindromes (1/2)

Soient $\Sigma$ un alphabet et $\mathcal{L} \in \Sigma^*$ tels que :

$$\begin{aligned} u \in \mathcal{L} \Leftrightarrow \quad & u = \varepsilon \\ \vee \quad & u = x \text{ avec } x \in \Sigma \\ \vee \quad & u = xvx \text{ avec } x \in \Sigma \wedge v \in \mathcal{L} \end{aligned}$$

Question : montrer que $\mathcal{L}$ est le langage des palindromes sur $\Sigma$

Preuve : par récurrence sur la longueur $n$ des mots

Hypothèse de récurrence : tout mot $u$ de $\mathcal{L}$ tel que $|u| \leq n$ est un palindrome et tout palindrome de taille inférieure ou égale à $n$ appartient à $\mathcal{L}$

Cas initiaux

- Si $n = 0$ alors $\varepsilon \in \mathcal{L}$ et $\varepsilon$ est un palindrome
- Si $n = 1$ alors tout mot $u = x$ avec $x \in \Sigma$, $u$ est un palindrome sur $\Sigma$ et $u \in \mathcal{L}$

# Démonstration « simple » par récurrence : palindromes (2/2)

Cas général

Soit $u \in \mathcal{L}$, tel que $|u| = n+1, n > 1$ alors $u = xvx$ avec $x \in \Sigma \wedge v \in \mathcal{L}$, comme $|v| = n-1$, par hypothèse de récurrence $v$ est un palindrome, et donc par définition, $u$ est un palindrome

Soit $u$ un palindrome tel que $|u| = n+1, n > 1$. Posons $u = x_0 x_1 \dots x_{n-1} x_n$ avec pour tout $0 \leq i \leq n$, $x_i \in \Sigma$. Par définition des palindromes $u = u^R = x_n x_{n-1} \dots x_1 x_0$ donc $x_0 = x_n$ et $u = x_0 v x_0$ avec $x_0 \in \Sigma \wedge v$ un palindrome. Comme $|v| = n-1$, par hypothèse de récurrence, $v \in \mathcal{L}$ et donc $u \in \mathcal{L}$

☛L'exemple semble trivial... effectivement, il l'est mais...

☛Attention, les erreurs de raisonnement sont faciles, surtout lorsque celui-ci parait simple

# Démonstration « fausse » par récurrence

Montrons par récurrence que les chaussettes d'un tiroir ont toute la même couleur (c'est plus facile pour éviter les erreurs le matin ☺)

- pour $n = 1$ la propriété est vraie
- hypothèse de récurrence : la propriété est vraie pour les tiroirs contenant un nombre de chaussettes $< n$
- dans un tiroir de $n$ chaussettes, faisons un tas avec les $n - 1$ premières chaussettes, par hypothèse de récurrence elles sont toutes de la même couleur. Faisons un tas avec les $n - 1$ dernières chaussettes, par hypothèse de récurrence elles sont toutes de la même couleur. Il y a au moins une chaussette en commun entre les deux tas de chaussettes, donc par transitivité, les $n$ chaussettes ont la même couleur

Quelle est l'erreur ?

Le raisonnement est faux pour $n = 2$ : l'hypothèse est vraie pour $n - 1 = 1$ mais il n'y a pas de chaussette en commun dans les 2 tas

# Démonstration par récurrence : autre exemple

Question Soient $u, v \in \Sigma^*$, montrer que $(uv)^R = v^R u^R$

Par induction sur la longueur de $v$.

Cas initial. Si $|v| = 0$, alors $v = \varepsilon$,
$(uv)^R = (u\varepsilon)^R = u^R = \varepsilon u^R = \varepsilon^R u^R = v^R u^R$

Hypothèse de récurrence Si $|v| \leq n$ alors $(uv)^R = v^R u^R$

Récurrence Soit $|v| = n + 1$, alors il existe $v = wa$ avec $w \in \Sigma^*$, $a \in \Sigma$ et $|w| = n$

$$
\begin{array}{rcll}
(uv)^R & = & (u(wa))^R & \text{car } v = wa \\
& = & ((uw)a)^R & \text{car la concaténation est associative} \\
& = & a(uw)^R & \text{par définition de l'inverse} \\
& = & aw^R u^R & \text{par hypothèse de récurrence} \\
& = & (wa)^R u^R & \text{par défintion de l'inverse} \\
& = & (v)^R u^R & \text{car } v = wa
\end{array}
$$

# Langages et expressions rationnelles

# De quoi parlons nous ?

D'une famille de langages que l'on peut décrire de manière très simple par des « motifs » ou « patterns »

Tous les langages finis sont rationnels mais également de nombreux langages infinis

S'il s'agit d'une des classes de langages les « plus simples », elle est tout de même très utile, par exemple pour la recherche de motifs dans des textes comme avec la commande `grep`

# Langage rationnel

L'union, la concaténation et l'étoile sont les **opérations rationnelles** à partir desquelles sont définis les langages rationnels

Définition Soit $\Sigma$ un alphabet, l'ensemble $Rat(\Sigma^*)$ des langages rationnels sur $\Sigma$ est le plus petit ensemble de langages tel que :

- $\emptyset \in Rat(\Sigma^*)$, $\{\varepsilon\} \in Rat(\Sigma^*)$
  ➥ contenant les langages vide et composé du mot vide
- $\forall x \in \Sigma, \{x\} \in Rat(\Sigma^*)$
  ➥ contenant les singletons de mots de une lettre de l'alphabet $\Sigma$
- $\forall \mathcal{L}_1 \in Rat(\Sigma^*), \forall \mathcal{L}_2 \in Rat(\Sigma^*)$, $\mathcal{L}_1 \cup \mathcal{L}_2 \in Rat(\Sigma^*)$
  $\mathcal{L}_1.\mathcal{L}_2 \in Rat(\Sigma^*)$
  $\mathcal{L}_1^* \in Rat(\Sigma^*)$
  ➥ clos par les opérations rationnelles

# Expressions rationnelles

Une expression rationnelle est un moyen de décrire de manière constructive un langage rationnel

Soit $\Sigma$ un alphabet. Les **expressions rationnelles** sur $\Sigma$ sont les mots de longueur finie définis inductivement par :

- $\emptyset$, $\varepsilon$ et $x$ pour tout $x \in \Sigma$ sont des expressions rationnelles
- si $e_1$ et $e_2$ sont des expressions rationnelles alors $(e_1 + e_2)$, $(e_1 e_2)$ et $(e_1)^*$ sont des expressions rationnelles

La priorité des opérateurs permet de supprimer les parenthèses pour simplifier l'écriture dans l'ordre : $*$, concaténation, $+$

# Exemples

$\Sigma = \{a, b, c, d\}$ :

- $(a + bc^*)d$ est une expression rationnelle
- $a + (bc^*d)$ est une expression rationnelle
- $(a + b + c)^* a (a + b + c)^* a (a + b + c)^*$ est une expression rationnelle
- $(b + c)^* a (b + c)^* a (a + b + c)^*$ est une expression rationnelle

# Expressions et langages

Une expression rationnelle $e$ représente un (unique) langage $\mathcal{L}(e)$, défini de manière inductive :

- $\mathcal{L}(\emptyset) = \emptyset$
- $\mathcal{L}(\varepsilon) = \{\varepsilon\}$
- $\forall x \in \Sigma, \mathcal{L}(x) = \{x\}$
- $\mathcal{L}(e_1 + e_2) = \mathcal{L}(e_1) \cup \mathcal{L}(e_2)$
- $\mathcal{L}(e_1 e_2) = \mathcal{L}(e_1)\mathcal{L}(e_2)$
- $\mathcal{L}(e_1^*) = \mathcal{L}(e_1)^*$
- $\mathcal{L}((e_1)) = \mathcal{L}(e_1)$

# Exemples

- $\mathcal{L}((a+bc^*)d) = \{ad, bd, bcd, bccd, bcccd, bccccd, \dots\}$
- $\mathcal{L}(a+(bc^*d)) = \{a, bd, bcd, bccd, bcccd, bccccd, \dots\}$
- $\mathcal{L}((a+b+c)^*a(a+b+c)^*a(a+b+c)^*)$ est le langage dont les mots commencent par un nombre quelconque de $a, b, c$ dans un ordre quelconque, suivi par un $a$, suivi par un nombre quelconque de $a, b, c$ dans un ordre quelconque, suivi par un nouveau $a$, suivi par un nombre quelconque de $a, b, c$ dans un ordre quelconque
- $\mathcal{L}((b+c)^*a(b+c)^*a(a+b+c)^*)$ est le langage dont les mots commencent par un nombre quelconque de $b, c$ dans un ordre quelconque, suivi par un $a$, suivi par un nombre quelconque de $b, c$ dans un ordre quelconque, suivi par un nouveau $a$, suivi par un nombre quelconque de $a, b, c$ dans un ordre quelconque

☛Les deux dernières expressions décrivent toutes deux le langages des mots sur $\{a, b, c\}$ qui contiennent au moins 2 $a$

# Équivalence

Une expression rationnelle représente un unique langage mais un langage peut être représenté par plusieurs expressions

Deux expressions rationnelles sont équivalentes si elles représentent le même langage

Les expressions rationnelles représentent une classe intéressante de langages mais certains langages assez simples ne sont pas rationnels par exemple :

$$\{a^n b^n \mid n \in \mathbb{N}\}$$