CTD 1: Langage rationnels

BUT 2 – Automates et Langages – R4.A12

1 Langages formels

1.1 Mots

Définition 1.1 (Alphabet et mots)**.** Un **alphabet** Σ est un ensemble fini de **symboles** (ou encore *lettres*). Un **mot** m sur un alphabet Σ est une suite finie $a_1 \dots a_n$ de symboles $a_i \in \Sigma$. La suite vide de symboles, notée ϵ, est appelée **mot vide**.

Les entiers naturels (par exemple 10, 777, 153) sont des mots construits sur l'alphabet $\Sigma_{\mathbb{N}} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$. Pour construire les mots représentant les entiers relatifs, il faut ajouter à l'alphabet le symbole de soustractions : $\Sigma_{\mathbb{Z}} = \Sigma_{\mathbb{N}} \cup \{-\}$. Les mots clefs du langage Java tels que `class`, `for` ou `Integer` sont construits sur l'alphabet $\Sigma_{\text{Java}} = \{\texttt{a}, \texttt{b}, \texttt{c}, \dots, \texttt{z}, \texttt{A}, \texttt{B}, \texttt{C}, \dots, \texttt{Z}\}$.

Exercice 1.
1. Quel alphabet permet de construire les mots représentant les nombres décimaux ?
2. Quel alphabet permet de construire les mots représentant les réels ?
3. Quel alphabet permet de construire les mots représentant les nombres rationnels ?

Définition 1.2. La **longueur** d'un mot m, notée $|m|$, est le nombre de symboles de m (en comptant les éventuelles répétitions).

Pour tout alphabet, ϵ est l'unique mot de longueur nulle ($|\epsilon| = 0$). Aussi, $|10| = 2$, $|\texttt{Integer}| = 7$.

Définition 1.3 (Concaténation)**.** Soient $m_1 = a_1 \dots a_k$ et $m_2 = b_1 \dots b_\ell$ deux mots sur Σ, la **concaténation** des mots m_1 et m_2, notée $m_1 \cdot m_2$ (ou simplement $m_1 m_2$), est la suite de symboles $a_1 \dots a_k b_1 \dots b_\ell$.

Ainsi, la concaténation de mots ja et va est le mot $java$.

Propriétés 1.1. L'opérateur de concaténation est
- associatif : $(m_1 \cdot m_2) \cdot m_3 = m_1 \cdot (m_2 \cdot m_3)$
- possède ϵ pour élément neutre : $\epsilon \cdot m = m \cdot \epsilon = m$
- n'est pas commutatif : en général $m_1 \cdot m_2 \neq m_2 \cdot m_1$

Définition 1.4 (Sous-mots et facteurs)**.**
- Le mot v est **sous-mot** du mot u si $u = a_1 \dots a_n$ et $v = a_{i_1} \dots a_{i_k}$ avec $\{i_1, \dots, i_k\} \subseteq \{1, \dots, n\}$ et $i_1 < \dots < i_k$.
- v est **facteur** de u s'il existe u_1 et u_2 tels que $u = u_1 \cdot v \cdot u_2$.
- Si $u_1 = \epsilon$ alors v est un **facteur gauche**, ou **préfixe** de u. De plus, si $v \neq u$ et $v \neq \epsilon$, c'est un préfixe **propre**.
- Si $u_2 = \epsilon$ alors v est un **facteur droit**, ou **suffixe** de u. De plus, si $v \neq u$ et $v \neq \epsilon$, c'est un suffixe **propre**.
- Si $v \neq \epsilon$ et, $u_1 \neq \epsilon$ ou $u_2 \neq \epsilon$ alors v est un **facteur propre** de u. De façon équivalente, v est facteur propre de u si $v \neq \epsilon$ et $v \neq u$.

Exercice 2. Donner tous les sous-mots et tous les facteurs (préfixe/suffixe/propre) des mots suivants sur l'alphabet $\Sigma = \{a, b, c\}$:

1. ϵ
2. a
3. abc

1.2 Langages

Définition 1.5 (Langage)**.** Un **langage** sur un alphabet Σ est un ensemble de mots sur Σ.

Les langages peuvent être finis ou infinis. Par exemple, sur l'alphabet $\Sigma_{\mathbb{B}} = \{0, 1\}$, le langage $L_1 = \{00, 01001, 1100\}$ est fini alors que le langage $L_2 = \{m \mid m \text{ représente un multiple de 3 en base 2}\}$ est infini. Attention : l'ensemble vide $\emptyset$ et le singleton $\{\epsilon\}$ sont des langages. De plus, ces langages sont différents !

Définition 1.6 (Concaténation de langages)**.** Si L_1 et L_2 sont deux langages sur Σ, la **concaténation** de L_1 et L_2 est le langage

$$L_1 \cdot L_2 = \{m_1 \cdot m_2 \mid m_1 \in L_1 \text{ et } m_2 \in L_2\}$$

Exercice 3. Soit l'alphabet $\Sigma = \{a, b, c\}$, et soient les langages $L_1 = \{cab, ab\}$ et $L_2 = \{ab, ba, \epsilon\}$. Déterminez les langages résultant des opérations suivantes :

1. $L_1 \cap L_2$
2. $L_1 \cup L_2$
3. $L_1 \backslash L_2$
4. $L_2 \backslash L_1$
5. $L_1 \cdot L_2$
6. $L_2 \cdot L_1$

Définition 1.7 (Puissance, étoile de Kleene et complémentaire)**.** Soit L un langage sur Σ, on définit

— $L^0 = \{\epsilon\}$

— $L^{i+1} = L^i \cdot L$, pour $i \geq 0$

— $L^\star = \bigcup_{i \geq 0} L^i$ (**étoile de Kleene**)

— $L^+ = L \cdot L^\star = \bigcup_{i > 0} L^i$

Le **complémentaire** d'un langage L sur Σ est le langage $\bar{L} = \{m \mid m \in \Sigma^\star \text{ et } m \notin L\}$.

Exercice 4. Pour les langages de l'exercice 3, déterminez les langages résultant des opérations suivantes :

1. L_1^0
2. L_2^0
3. L_1^2
4. L_2^2
5. Σ^*

Exercice 5. Soit le langage $L = \{a\}$ composé de l'unique mot a.

1. Quel est le nombre de mots du langage L^{50} ?
2. Quel est le nombre de mots du langage L^i , pour $i \geq 1$?

3. Quel est le nombre $||P(L)||$ de mots du langage

$$P(L) = \bigcup_{1 \leq i \leq 50} L^i = L \cup L^2 \dots L^{50} ?$$

4. Quel est le nombre $||E(L)||$ de mots du langage

$$E(L) = \bigcup_{0 \leq i \leq 50} L^i = \{\epsilon\} \cup L \cup L^2 \dots L^{50} ?$$

5. Répondre aux mêmes questions pour les langages $L_1 = \{\epsilon, a\}$, $L_2 = \{aa, a\}$ et $L_3 = \{aa, b\}$.

Exercice 6. Soient l'alphabet $\Sigma = \{a, b\}$ et les langages suivants : $L_1 = \{a^n b^p : n, p \in \mathbb{N}\}$, $L_2 = \{a^n b^n : n \in \mathbb{N}\}$, $L_3 = \{a^n : n \in \mathbb{N}\} = a^\star$, $L_4 = \{b^n : n \in \mathbb{N}\} = b^\star$. On rappelle que, par convention, on pose $a^0 = b^0 = \epsilon$.

1. Donnez trois mots de chacun de ces langages.
2. Déterminez $L_1 \cap L_2$.
3. Déterminez $L_1 \backslash L_2$, puis $L_2 \backslash L_4$.
4. Les égalités suivantes sont-elles valides ?
   — $L_1 = L_3 \cdot L_4$;
   — $L_2 = L_3 \cdot L_4$

2 Langages rationnels

Définition 2.1 (Langages rationnels)**.** Soit Σ un alphabet, l'**ensemble $\mathcal{R}at$ des langages rationnels** sur Σ est *le plus petit* ensemble de langages satisfaisant les conditions suivantes :

— $\emptyset \in \mathcal{R}at, \{\epsilon\} \in \mathcal{R}at$;
— $\{a\} \in \mathcal{R}at$ pour tout $a \in \Sigma$;
— si $L_1, L_2 \in \mathcal{R}at$ alors $L_1 \cup L_2 \in \mathcal{R}at$, $L_1 \cdot L_2 \in \mathcal{R}at$ et $L^\star \in \mathcal{R}at$.

Dit autrement, l'ensemble des langages rationnels sur un alphabet Σ, est le plus petit ensemble de langages qui contient le langage vide $\emptyset$, les langages singleton $\{a\}$ pour tout $a \in \Sigma$, le langage $\{\epsilon\}$ et qui est clos par union, concaténation et étoile de Kleene. Le langage

$$L = \{\epsilon, ab, abab, ababab, \dots\}$$

est un langage rationnel. En effet, étant donnés $L_0 = \{a\}$ et $L_1 = \{b\}$ (des langages, par définition, rationnels), on peut facilement voir que $L = (L_0 \cdot L_1)^\star$ et donc L est rationnel.

Définition 2.2 (Expressions rationnelles)**.** Les **expressions rationnelles** sur un alphabet Σ sont les expressions formées par les règles suivantes :

— $\emptyset, \epsilon$ et les éléments de Σ sont des expressions rationnelles
— si e_1 et e_2 sont des expressions rationnelles, alors $e_1 + e_2$ et $e_1 \cdot e_2$ (ou simplement $e_1 e_2$) sont des expressions rationnelles
— si e est une expression rationnelle, alors (e) et $e^\star$ sont des expressions rationnelles.

Voici quelques exemples d'expressions rationnelles :

— $a^\star$

— $(a+b+c)^\star + d^\star$

— $ab^\star + c$

Dans ce dernier cas, pour interpréter correctement l'expression rationnelle, il convient de spécifier la priorité des opérateurs : $\star > \cdot > +$. Ainsi, $ab^\star + c = (a \cdot (b^\star)) + c$.

Définition 2.3. Le langage $\mathcal{L}(e)$ représenté par l'expression rationnelle e est définit ainsi :

— $\mathcal{L}(\emptyset) = \emptyset$, $\mathcal{L}(\epsilon) = \{\epsilon\}$

— $\mathcal{L}(a) = \{a\}$

— $\mathcal{L}(e_1 + e_2) = \mathcal{L}(e_1) \cup \mathcal{L}(e_2)$

— $\mathcal{L}(e_1 \cdot e_2) = \mathcal{L}(e_1) \cdot \mathcal{L}(e_2)$

— $\mathcal{L}(e^\star) = \mathcal{L}(e)^\star$

Par exemple, $\mathcal{L}(a^\star b + b^\star a) = \mathcal{L}(a^\star b) \cup \mathcal{L}(b^\star a) = \mathcal{L}(\{a\})^\star \cdot \mathcal{L}(\{b\}) \cup \mathcal{L}(\{b\})^\star \cdot \mathcal{L}(\{a\}) = \{a^n \mid n \in \mathbb{N}\} \cdot \{b\} \cup \{b^n \mid n \in \mathbb{N}\} \cdot \{a\}$.

Théorème 2.1. Un langage est rationnel si et seulement s'il existe une expression rationnelle qui le représente.

Exercice 7. Donnez une expression rationnelle des langages suivants sur l'alphabet romain $\Sigma = \{a, \ldots, z\}$.

1. les mots commençant par a
2. les mots finissant par b
3. les mots ayant un nombre pair de a
4. les mots ayant un nombre impair de voyelles
5. les mots ayant un nombre impair de voyelles et commençant par a

Exercice 8. En java, le type `double` permet de représenter les nombres à virgule flottante. Un littéral double peut contenir un point (symbolisant la virgule), mais ce n'est pas obligatoire. Lorsqu'il possède un point, il y a forcément un nombre avant ou après le point. Voici quelques exemples de littéraux double :

```
0.4    .4    10    10.4    10.
```

Un littéral double peut posséder un signe (`+` ou `-`). Il peut aussi avoir un facteur d'échelle précédé par `e` (ou `E`) qui signifie ** 10 puissance ...*, ce facteur d'échelle étant un entier signé. Voici d'autres exemples de littéraux double :

```
3.86473e5    123.0e+8   -12E4  .45E-23   -16    012.e78
```

Donnez une expression rationnelle pour représenter les littéraux double en JAVA. On pourra associer un nom à des sous-expressions, comme *chiffre*, *nombre*,... afin d'obtenir une expression plus lisible.

Exercice 9. On considère l'alphabet $\Sigma_{\mathbb{B}} = \{0, 1\}$.

1. Donnez une expression rationnelle représentant l'ensemble des mots qui sont des multiples de 2 en base 2.
2. Donnez une expression rationnelle représentant l'ensemble des mots qui ont la même longueur que leur successeur en base 2.

3. Donnez une expression rationnelle représentant l'ensemble des mots qui n'ont pas deux chiffres identiques successifs.

Exercice 10. Pour chacune des expressions rationnelles suivantes sur l'alphabet $\Sigma = \{a, b\}$ donnez une expression rationnelle représentant son complémentaire :

1. $(a + b)^* b$
2. $((a + b)(a + b))^*$
3. $(a + b)^* a(a + b)^*$
4. $(a + b)^* aa(a + b)^*$