I Alphabet et mot

Définition : Alphabet

Définition : Mot

Le mot vide (contenant aucune lettre) est noté ε.
Σ^* est l'ensemble des mots sur Σ. $\Sigma^+ = \Sigma^* \setminus \{\varepsilon\}$. Σ^n est l'ensemble des mots de longueur n sur Σ.

Définition : Égalité de mots

Deux mots $u = u_1...u_n$ et $v = v_1...v_p$ sur le même alphabet Σ sont égaux s'ils ont la même longueur ($n = p$) et si pour tout $i \in \{1, ..., n\}$, $u_i = v_i$.

Définition : Concaténation et puissance

Exercice 1.
Soient Σ un alphabet, a, $b \in \Sigma$ et $u \in \Sigma^*$. On suppose $au = ub$.
Montrer que $a = b$ et qu'il existe $k \in \mathbb{N}$ tel que $u = a^k$.

Définition : Préfixe, suffixe, facteur, sous-mot

- u est un préfixe de m s'il existe un mot v tel que $m = uv$.
- u est un suffixe de m s'il existe un mot v tel que $m = vu$.
- u est un facteur (*substring* en anglais) de m s'il existe des mots v, w tels que $m = vuw$.
- u est un sous-mot (*subsequence* en anglais) de m si u est une sous-suite (ou : suite extraite) de m.

Exemple : abc est un sous-mot de $aabacb$, mais pas un facteur.

II Langage

Définition : Langage

Exemples :

1. L'ensemble L_1 des mots du dictionnaire français sur $\Sigma = \{a, b, ..., z\}$.
2. L'ensemble L_2 des formules arithmétiques sur $\Sigma = \{0, ..., 9, +, -, /, *\}$.
3. L'ensemble L_3 des programmes OCaml sur $\Sigma = \{a, ..., z, , !, <, >, ...\}$.
4. L'ensemble L_4 des ADN sur $\Sigma = \{A, C, G, T\}$.

Définition : Concaténation

Définition : Puissance

Exemple : Σ^n est l'ensemble des mots de longueur n sur l'alphabet Σ.

Exercice 2.
Soit L un langage.

1. À quelle condition a t-on $L \subseteq L^2$?
2. Quel lien a t-on entre L^2 et $\{u^2 \mid u \in L\}$?

Définition : Étoile de Kleene

Remarque : L^* contient toujours ε car $L^0 = \{\varepsilon\}$.

Exercice 3.
Montrer que $(L^*)^* = L^*$.

III Langages réguliers

Définition : Langage régulier (ou : langage rationnel)

Définition inductive équivalente :

Propriété

- Tout langage fini est régulier
- L_1 et L_2 réguliers $\implies$ $L_1 \cup L_2$ régulier
- L_1 et L_2 réguliers $\implies$ $L_1 L_2$ régulier
- L régulier $\implies$ L^* régulier

Par récurrence immédiate, si L_1, ..., L_n sont réguliers alors $L_1 \cup ... \cup L_n$ et $L_1 L_2 ... L_n$ sont réguliers.
Attention : une union infinie de langages réguliers n'est pas forcément régulière.
Exemples :

1. Soit m un mot. Alors $\{m\}$ est fini donc est un langage régulier, qu'on note aussi m par abus de langage.
2. Σ est fini donc est régulier. Σ^* est l'étoile d'un langage régulier donc est régulier.
3. Soit m un mot. L'ensemble des mots ayant comme facteur m est égal à $\Sigma^* m \Sigma^*$ donc est un langage régulier.
4. Soit $m = m_1 \cdots m_n$ un mot. L'ensemble des mots ayant comme sous-mot m est égal à $\Sigma^* m_1 \Sigma^* m_2 \cdots \Sigma^* m_n \Sigma^*$ donc est un langage régulier.

Exercice 4.
Montrer que les langages suivants sont réguliers sur $\Sigma = \{a, b\}$:

1. Mots commençants par a :
2. Mots commençants par a et finissant par b :
3. Mots de taille paire :
4. Mots de taille impaire :

IV Expressions régulières

Les expressions régulières sont une notation plus concise pour représenter un langage régulier :

Définition : Expression régulière (ou : expression rationnelle)

L'ensemble des expressions régulières sur un alphabet Σ est le plus petit langage $\mathcal{R}$ sur $\Sigma \cup \{\emptyset, \varepsilon, |, ^*, (,)\}$ vérifiant :

- $\forall a \in \Sigma$, $a \in \mathcal{R}$
- $\emptyset \in \mathcal{R}$, $\varepsilon \in \mathcal{R}$
- $\forall e_1, e_2 \in \mathcal{R}$, $(e_1 | e_2) \in \mathcal{R}$ et $(e_1 e_2) \in \mathcal{R}$
- $\forall e \in \mathcal{R}$, $e^* \in \mathcal{R}$

On peut les représenter informatiquement par le type OCaml :

```ocaml
type 'a regexp =
| Vide | Epsilon | L of 'a (* L a est la lettre a *)
| Union of 'a regexp * 'a regexp
| Concat of 'a regexp * 'a regexp
| Etoile of 'a regexp
```

Définition : Langage d'une expression régulière

Si e est une expression régulière, on définit le langage $L(e)$ récursivement :
- $L(a) = \{a\}$ si $a \in \Sigma$
- $L(\emptyset) = \emptyset$, $L(\varepsilon) = \{\varepsilon\}$
- $L(e|e') = L(e) \cup L(e')$
- $L(ee') = L(e)L(e')$
- $L(e^*) = L(e)^*$

Remarques :

- Par abus de langage, on confond souvent e et $L(e)$.
- Les expressions rationnelles sont une façon plus pratique de décrire les langages réguliers, en utilisant $|$ au lieu de $\cup$ et en omettant les parenthèses.

Théorème

Soit L un langage.
L est régulier si et seulement si il existe une expression régulière e telle que $L = L(e)$.

Exemples :

- $(a|b)^*$: ensemble de tous les mots $(= \Sigma^*)$.
- $(a|b)^*bb$: mots finissant par bb.

Exercice 5.
Donner une expression régulière pour les langages suivants, sur $\Sigma = \{a, b\}$:

1. Mots contenant au plus un a : ____________________
2. Mots de taille $n \equiv 1 \mod 3$: ____________________
3. Mots contenant un nombre pair de a : ____________________
4. Mots contenant un nombre impair de a : ____________________
5. Écritures en base 2 des entiers divisibles par 4 : ____________________

V Induction structurelle

Théorème : Induction structurelle sur les langages réguliers

Soit $\mathcal{P}(L)$ une propriété sur les langages réguliers L telle que :
- $\mathcal{P}(L)$ est vraie pour les langages L finis (cas de base)
- $\mathcal{P}(L_1) \wedge \mathcal{P}(L_2) \implies \mathcal{P}(L_1L_2)$
- $\mathcal{P}(L_1) \wedge \mathcal{P}(L_2) \implies \mathcal{P}(L_1 \cup L_2)$
- $\mathcal{P}(L) \implies \mathcal{P}(L^*)$

Alors $\mathcal{P}(L)$ est vraie pour tout langage régulier L.

Preuve : ____________________

De même pour les expressions régulières :

Théorème : Induction structurelle sur les expressions régulières

Soit $\mathcal{P}(e)$ une propriété sur les expressions régulières telle que :

- $\mathcal{P}(\emptyset)$, $\mathcal{P}(\varepsilon)$ sont vraies (cas de base)
- $\mathcal{P}(a)$ est vraie pour $a \in \Sigma$ (cas de base)
- $\mathcal{P}(e_1) \land \mathcal{P}(e_2) \implies \mathcal{P}(e_1 e_2)$
- $\mathcal{P}(e_1) \land \mathcal{P}(e_2) \implies \mathcal{P}(e_1 \cup e_2)$
- $\mathcal{P}(e) \implies \mathcal{P}(e^*)$

Alors $\mathcal{P}(e)$ est vraie pour toute expression régulière e.

Exercice 6.
Si $m = m_1 ... m_n$ est un mot, on définit son miroir $\widetilde{m} = m_n ... m_1$.
Si L est un langage, on définit son miroir $\widetilde{L} = \{\widetilde{m} \mid m \in L\}$.

1. Donner une expression régulière du miroir de $a(a|b)^*b$.
2. Soit e une expression régulière de langage L. Montrer que $\widetilde{L}$ est régulier.
3. Écrire une fonction Caml `miroir : 'a regexp -> 'a regexp` renvoyant le miroir d'une expression régulière.