

Exercice 1. Mots

1. Soit $\Sigma = \{a, b\}$. Déterminer tous les mots sur Σ dont tous les facteurs sont des préfixes.
2. (a) Soient u et v deux mots ayant le même ensemble de préfixes. u et v sont-ils nécessairement égaux ?
 (b) Même question pour l'ensemble des préfixes propres (ensemble des préfixes différents du mot entier).
3. Soient a, b deux lettres et u un mot tel que $au = ub$. Montrer que u est d'une certaine forme qu'on précisera.
4. (Lemme de Levi) Soient $u, v, u', v' \in \Sigma^*$ des mots tels que $uv = u'v'$. Montrer qu'il existe un unique mot w tel que :

$$\begin{cases} u = u'w \\ v' = wv \end{cases} \quad \text{ou} \quad \begin{cases} u' = uw \\ v = wv' \end{cases}$$

Exercice 2. Langages

1. Soit L un langage non vide. Montrer que $\epsilon \in L$ si et seulement si $L \subset L \cdot L$.
2. Pour tout langage L , on note $\text{pref}(L)$ l'ensemble des préfixes des mots de L . Soient L_1 et L_2 deux langages. Déterminer $\text{pref}(L_1 \cdot L_2)$
3. (Lemme d'Arden) Soit Σ un alphabet et A, B deux langages tels que $\epsilon \notin A$. Résoudre l'équation $L = AL \cup B$ d'inconnue $L \subset \Sigma^*$.

Indication : on pourra utiliser la même méthode que dans le cours où on traité le cas particulier $A = \{a\}$ et $B = \{b\}$.

Exercice 3. Mots de Fibonacci

Soit $\Sigma = \{a, b\}$. Les mots de Fibonacci sont définis par récurrence :

$$\begin{cases} f_0 = \epsilon, f_1 = a, f_2 = b \\ \forall n \geq 1 : f_{n+2} = f_{n+1} \cdot f_n \end{cases}$$

1. Soit $n \geq 3$. Déterminer les deux dernière lettres de f_n . Justifier.
2. Pour tout $n \geq 3$, on note g_n le mot f_n duquel on a supprimé les deux dernière lettres. Montrer que g_n est un palindrome.

Exercice 4.

Soit u et v deux mots non vides. Montrer que :

$$\begin{array}{ccc} \text{Il existe deux mots } x, y \text{ tels que} & \Leftrightarrow & \text{Il existe un mot } w \text{ tel que} \\ u = x \cdot y \text{ et } v = y \cdot x & & u \cdot w = w \cdot v. \end{array}$$

Exercice 5. Code sur un alphabet

Soit Σ un alphabet et $L \subset \Sigma^*$ un langage. On dit que L est un **code** sur Σ lorsque :

Pour tout $(p, q) \in \mathbb{N}^2$, pour tout $x_1, \dots, x_p, y_1, \dots, y_q \in L$,
 si $x_1 \dots x_p = y_1 \dots y_q$, alors $p = q$ et $x_1 = y_1, \dots, x_p = y_p$.

1. Parmi les langages suivants, déterminer lesquels sont des codes :

$$L_1 = \{a, ba, bba, baab\}$$

$$L_2 = \{aa, ab, aab, bba\}$$

$$L_3 = \{ab, baa, abba, aabaa\}$$

$$L_4 = \{b, ab, baa, abaa, aaaa\}$$

2. Montrer que si $\epsilon \in L$, alors L n'est pas un code.
3. Soit $u \in \Sigma^*$ un mot non vide. Montrer que $\{u\}$ est un code.
4. Soient u et v deux mots distincts non vides. Montrer que $\{u, v\}$ est un code si et seulement si $uv \neq vu$.

Exercice 6. Mots de Bruijn

Soit Σ un alphabet et $u = u_0 \dots u_{n-1}$ un mot non vide.

- ★ Soit $i \in \llbracket 0, n-1 \rrbracket$ et $k \in \mathbb{N}^*$. Le **facteur circulaire d'indice i et de longueur k** de u est le mot :

$$u_{r(i)}u_{r(i+1)}u_{r(i+2)} \dots u_{r(i+k-1)}$$

où $r(j)$ est le reste dans la division euclidienne de j par n .

- ★ Soit $k \in \mathbb{N}^*$. Le mot $u = u_0 \dots u_{n-1}$ est un **mot de Bruijn d'ordre k** si pour tout $v \in \Sigma^k$, il existe un unique $i \in \llbracket 0, n-1 \rrbracket$ tel que v est le facteur circulaire d'indice i et de longueur k de u .

1. Déterminer un mot de Bruijn pour $\Sigma = \{a, b\}$ et $k \in \{1, 2, 3\}$.
2. Si u est un mot de Bruijn d'ordre k , que vaut $|u|$?

Pour $k \in \mathbb{N}^*$, on note $G_k = (\Sigma^k, E_k)$ le graphe orienté dans lequel il existe un arc de u vers v si et seulement si le suffixe de u de longueur $k-1$ est un préfixe de v .

Théorème (Théorème d'Euler). *Soit G un graphe orienté fortement connexe (c'est à dire que pour tout couple de sommets (u, v) , il existe un chemin de u vers v et inversement). Alors :*

G admet un cycle passant une et une seule fois par chaque arc

\Leftrightarrow

Pour tout sommet u , le degré entrant de u est égal au degré sortant de u .

3. Expliquer comment utiliser le graphe G_k pour construire un mot de Bruijn d'ordre $k+1$.
4. Donner un mot de Bruijn d'ordre 4 pour $\Sigma = \{a, b\}$ et d'ordre 2 pour $\Sigma = \{a, b, c\}$.

Algorithme de Hierholzer. Soit $G = (S, A)$ un graphe orienté admettant un cycle passant une et une seule fois par chaque arc. Pour construire un tel cycle :

- On choisit un sommet initial $s_0 \in S$ et on suit les arcs de G , en s'interdisant de repasser deux fois par le même arc. Comme chaque sommet a le même degré entrant et sortant, lorsque l'algorithme ne peut plus emprunter d'arcs, le dernier sommet visité est nécessairement s_0 .
 On obtient donc un cycle γ entre s_0 et lui-même, mais qui n'emprunte pas nécessairement tous les arcs de G .
 - Tant qu'il existe un sommet s de γ ayant un arc sortant n'ayant pas été emprunté, on construit comme dans l'étape précédente un cycle γ' entre s et lui-même sans emprunter les arcs de γ .
 - Soient γ_1 et γ_2 les chemins tels que $\gamma = \gamma_1 + (s_0) + \gamma_2$ où les $+$ représentent des concaténations de chemins. On recommence l'étape précédente en remplaçant γ par $\gamma_1 + \gamma' + \gamma_2$.
5. À l'aide de l'algorithme de Hierholzer, écrire une fonction OCaml qui renvoie un mot de Bruijn d'ordre $k \in \mathbb{N}^*$ pour l'alphabet $\Sigma = \llbracket 1; n \rrbracket$.