

**ECOLES NORMALES SUPERIEURES  
CONCOURS D'ADMISSION 2025**

**VENDREDI 18 AVRIL 2025  
14h00 - 18h00  
FILIERE MP etMPI - EPREUVE n° 10  
INFO-FONDAMENTALE (ULSR)**

Durée : 4 heures  
L'utilisation des calculatrices n'est pas autorisée pour cette épreuve

# Forêts de factorisation

Le sujet comporte 12 pages, numérotées de 1 à 12.

## Préliminaires

Étant donnés deux ensembles  $A$  et  $B$ , on note  $B^A$  l'ensemble des fonctions de  $A$  dans  $B$ .

Dans ce sujet, un **alphabet** est toujours un ensemble fini dont les éléments sont appelés des **lettres**. Étant donné un alphabet  $\Sigma$ , on notera  $\Sigma^*$  l'ensemble des mots finis sur  $\Sigma$ . On notera le mot vide par  $\varepsilon$ . On notera aussi  $\Sigma^+$  l'ensemble des mots finis **non vides** sur  $\Sigma$ . On notera  $u \cdot v$  la concaténation des mots  $u, v \in \Sigma^*$ . La longueur d'un mot  $u$  sera notée  $|u|$ .

On notera  $u[1], \dots, u[n]$  les lettres d'un mot  $u$  de longueur  $n$ . Pour tous indices  $1 \leq i \leq j \leq n$ , on notera  $u[i, \dots, j]$  le mot formé des lettres  $u[i], \dots, u[j]$ . On dit qu'un mot  $x$  est un **facteur** de  $u$  s'il existe des indices  $i$  et  $j$  tels que  $x = u[i, \dots, j]$ .

 On notera bien que dans ce sujet, les mots sont indicés à partir de 1 et que le facteur  $w[i, \dots, j]$  d'un mot  $w$  contient la lettre  $w[i]$  et la lettre  $w[j]$ .

**Exemple 1.** Sur l'alphabet  $\Sigma = \{\mathbf{a}, \mathbf{b}\}$ , le mot  $u = \mathbf{aaba}$  est de longueur  $|u| = 4$ . Ses lettres sont  $u[1] = \mathbf{a}$ ,  $u[2] = \mathbf{a}$ ,  $u[3] = \mathbf{b}$  et  $u[4] = \mathbf{a}$ . Le facteur  $u[1, \dots, 4]$  est le mot  $u$  lui-même, le facteur  $u[2, \dots, 3]$  est le mot  $\mathbf{ab}$  et le facteur  $u[2, \dots, 4]$  est le mot  $\mathbf{aba}$ .

Lorsque du code est demandé, on écrira la solution en pseudo-code ou dans un langage de programmation au choix de la candidate ou du candidat. On pourra s'inspirer de la syntaxe suivante, donnée à titre indicatif

---

### Fonction PremiersFibonacci( $n$ )

---

```
1 début
2   fib ← tableau de  $n$  entiers initialisés
3   fib[1] ← 1
4   fib[2] ← 1
5   pour tout les  $k$  allant de 3 à  $n$  faire
6     fib[k] ← fib[k - 2] + fib[k - 1]
7   renvoyer fib
```

---

Les parties I et II sont à traiter en premier. Les parties III, IV et V sont indépendantes et peuvent être traitées dans n'importe quel ordre.

## Partie I. Semi-groupes

Un **semi-groupe**  $(S, \cdot)$  est un magma associatif, c'est-à-dire un ensemble  $S$  muni d'une loi interne «  $\cdot$  » associative :  $a \cdot (b \cdot c) = (a \cdot b) \cdot c$  pour tous  $a, b, c \in S$ . Un semi-groupe  $(S, \cdot)$  est dit **fini** si l'ensemble  $S$  est fini. La **taille** d'un semi-groupe fini  $(S, \cdot)$  est le cardinal  $|S|$  de  $S$ . Dans la suite, on notera l'opération «  $\cdot$  » sous forme multiplicative, c'est-à-dire  $a \cdot b = ab$ . Notons que tout groupe est un semi-groupe. Voici d'autres exemples de semi-groupes.

**Exemple 2.** L'ensemble  $\mathbb{N}$  des entiers naturels muni de la multiplication est un semi-groupe. Ce même ensemble  $\mathbb{N}$  muni de l'addition est aussi un semi-groupe.

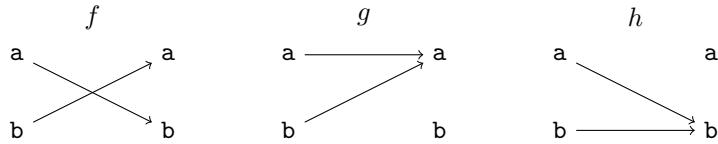
**Exemple 3.** Soit  $\Sigma$  un alphabet. L'opération de concaténation sur les mots est associative. Ainsi,  $(\Sigma^*, \cdot)$  et  $(\Sigma^+, \cdot)$  sont des semi-groupes. Dans la suite, ces semi-groupes seront notés  $\Sigma^*$  et  $\Sigma^+$  respectivement.

On dit qu'un élément  $x$  d'un semi-groupe  $S$  est **neutre** (ou **identité**) si  $xy = yx = y$  pour tout  $y \in S$ . On dit qu'un élément  $e$  d'un semi-groupe  $S$  est **idempotent** si  $e^2 = e$ . Tout élément identité est idempotent mais l'inverse n'est pas nécessairement vrai.

**Question 1.** Montrer que dans tout semi-groupe, s'il existe un élément neutre alors il est unique.

Soit  $X$  un ensemble. On note  $T(X)$  le semi-groupe  $(X^X, \circ)$  où  $\circ$  est définie pour tous  $f, g \in X^X$  par  $f \circ g = g \circ f$  (ainsi,  $f \circ g$  est la fonction qui à  $x \in X$  associe  $g(f(x)) \in X$ ). On prêtera attention à l'ordre de composition inversé par rapport à  $\circ$ . On appelle  $T(X)$  le **semi-groupe de transition** sur  $X$ .

**Exemple 4.** Soit  $X = \{a, b\}$ . On peut remarquer que  $T(X) = \{\text{id}, f, g, h\}$  est un semi-groupe, où  $\text{id}$  est la fonction identité,  $f(a) = b$  et  $f(b) = a$ ,  $g(a) = g(b) = a$ , et  $h(a) = h(b) = b$ .



**Question 2.** Montrer que dans l'exemple précédent,  $\text{id}$  est neutre, et  $g$  est idempotent mais pas neutre.

On note  $\mathbb{B}$  le sous-ensemble  $\{\text{id}, f\}$  de  $T(X)$ , où  $f$  et  $X$  sont définis comme dans l'exemple 4.

**Question 3.** Montrer que  $(\mathbb{B}, \circ)$  est un semi-groupe dont on donnera la table de multiplication.

Dans toute la suite, on identifiera  $\mathbb{B}$  avec l'ensemble des booléens  $\{0, 1\}$ , en identifiant « 0 » avec  $\text{id}$  et « 1 » avec  $f$ . On remarque alors que la loi interne  $\circ$  est l'opérateur « OU exclusif », noté  $\oplus$  et défini par  $1 \oplus 0 = 0 \oplus 1 = 1$  et  $0 \oplus 0 = 1 \oplus 1 = 0$ .

Soient  $(S, \cdot)$  et  $(T, \star)$  deux semi-groupes. Un **morphisme** de semi-groupe est une application  $\phi : S \rightarrow T$  qui préserve la loi interne :  $\phi(x \cdot y) = \phi(x) \star \phi(y)$  pour tous  $x, y \in S$ .

**Exemple 5.** Soit  $\Sigma = \{a, b\}$ . La fonction  $\Psi_a : \Sigma^* \rightarrow (\mathbb{B}, \oplus)$  est définie pour tout mot  $w \in \Sigma^*$  par

$$\Psi_a(\varepsilon) = 0,$$

$$\Psi_a(wa) = \Psi_a(w) \oplus 1,$$

$$\Psi_a(wb) = \Psi_a(w).$$

On admettra que  $\Psi_a$  est un morphisme de semi-groupe.

**Question 4.** Montrer que  $\Psi_{\mathbf{a}}(w) = 0$  si, et seulement si,  $w$  contient un nombre pair de  $\mathbf{a}$ .

Soit  $\mathcal{A} = (Q, q_0, \delta, F)$  un automate fini déterministe (AFD). On suppose sans perte de généralité que l'automate est complet, c'est-à-dire que la fonction de transition est de la forme  $\delta : Q \times \Sigma \rightarrow Q$ . On étend  $\delta$  en une fonction  $\delta^* : Q \times \Sigma^* \rightarrow Q$  définie récursivement par  $\delta^*(q, \varepsilon) = q$  et  $\delta^*(q, w\sigma) = \delta(\delta^*(q, w), \sigma)$  pour tous  $q \in Q$ ,  $w \in \Sigma^*$  et  $\sigma \in \Sigma$ . On admettra l'identité suivante : pour tous  $u, v \in \Sigma^*$  et  $q \in Q$ .

$$\delta^*(q, uv) = \delta^*(\delta^*(q, u), v) \quad (1)$$

On note  $L$  le langage reconnu par l'automate  $\mathcal{A}$ . On pose

$$T(\mathcal{A}) = T(Q), \quad F' = \{f \in Q^Q : f(q_0) \in F\}.$$

De plus, on définit une fonction  $\psi : \Sigma^* \rightarrow T(\mathcal{A})$  comme suit. Étant donné un mot  $w \in \Sigma^*$ , on définit  $\psi(w) \in T(\mathcal{A})$  comme étant la fonction de  $Q$  vers  $Q$  qui à  $q \in Q$  associe  $\delta^*(q, w)$ . Ainsi,  $\psi(w)(q) = \delta^*(q, w)$ .

**Question 5.** Montrer que  $\psi$  est un morphisme de semi-groupe de  $\Sigma^*$  dans  $T(\mathcal{A})$ .

**Question 6.** Montrer que  $\psi^{-1}(F') \subseteq L$ .

**Question 7.** Montrer que  $L \subseteq \psi^{-1}(F')$ .

On a donc  $L = \psi^{-1}(F')$ . Dans toute la suite, on admettra le résultat suivant qui établit une correspondance entre langages réguliers et semi-groupes finis.

**Lemme 1.** Soit  $\Sigma$  un alphabet. Un ensemble  $L \subseteq \Sigma^*$  est un langage régulier si et seulement s'il existe semi-groupe fini  $(S, \cdot)$ , un ensemble  $F \subseteq S$  et un morphisme de semi-groupe  $\phi : \Sigma^* \rightarrow S$  tels que  $\phi^{-1}(F) = L$ .

## Partie II. Forêts de factorisation

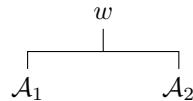
Soit  $\Sigma$  un alphabet,  $(S, \cdot)$  un semi-groupe et  $\phi : \Sigma^+ \rightarrow S$  un morphisme de semi-groupe.

Un **arbre de factorisation** d'un mot  $w \in \Sigma^+$  (pour  $\phi$ ) est un arbre étiqueté par  $\Sigma^+$  qui satisfait les conditions suivantes.

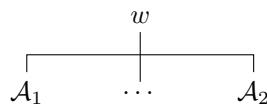
1. L'arbre de factorisation d'une lettre  $\sigma \in \Sigma$  est constitué d'une racine sans enfants étiquetée par cette lettre  $\sigma$  :

$\sigma$

2. Considérons un mot  $w = w_1 w_2$ , avec  $w_1, w_2 \in \Sigma^+$  non vides. Soient  $\mathcal{A}_1$  et  $\mathcal{A}_2$  des arbres de factorisation de  $w_1$  et  $w_2$  respectivement. Alors l'arbre dont la racine est étiquetée par  $w = w_1 w_2$  et qui possède deux enfants  $\mathcal{A}_1$  et  $\mathcal{A}_2$ , est un arbre de factorisation de  $w$  :



3. **Règle de l'idempotence** : Considérons un mot  $w = w_1 \dots w_n$ , avec  $w_1, \dots, w_n \in \Sigma^+$  non vides et  $n \geq 3$ . On suppose que  $\phi(w_1) = \dots = \phi(w_n)$  est un élément idempotent de  $S$ . Soient  $\mathcal{A}_1, \dots, \mathcal{A}_n$  des arbres de factorisation de  $w_1, \dots, w_n$  respectivement. Alors l'arbre dont la racine est  $w = w_1 \dots w_n$  et qui possède  $n$  enfants  $\mathcal{A}_1, \dots, \mathcal{A}_n$ , est un arbre de factorisation de  $w$  :



On prêtera attention à plusieurs points de cette définition :

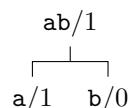
- on ne s'intéresse qu'aux mots non vides,
- il peut exister plusieurs arbres de factorisation pour un même mot  $w$  (voir exemples plus bas),
- l'ordre des enfants d'un nœud est important,
- lorsqu'un nœud possède trois enfants ou plus (dernier cas), il doit vérifier la règle de l'idempotence.

On utilisera les conventions graphiques suivantes :

- une **double barre horizontale** indiquera une application de la règle de l'idempotence (voir cas 2 de l'exemple 6 ci-dessous) ;
- lorsque l'on écrit un noeud  $w$  dans un arbre, on écrira  $w/\phi(w)$  afin de faciliter les raisonnements sur la règle de l'idempotence.

**Exemple 6.** Soit  $\Psi_a$  définie à l'exemple 5 page 2. On rappelle que  $\Psi_a(w) = 0$  si  $w$  contient un nombre pair de  $a$ , et  $\Psi_a(w) = 1$  sinon. On va donner des exemples d'arbres de factorisation pour  $\Psi_a$ . Regardons quelques exemples de mots :

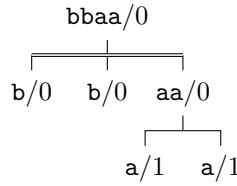
1.  $u = ab$  : on applique le deuxième cas de la définition avec  $w_1 = a$  et  $w_2 = b$ . Ainsi,  $w_1$  et  $w_2$  sont des lettres et  $u = w_1 w_2$ . En appliquant deux fois le premier cas de la définition, on obtient l'arbre de factorisation suivant pour  $u = ab$  :



On notera que  $\Psi_a(a) = 1$ ,  $\Psi_a(b) = 0$  et  $\Psi_a(ab) = 0$ .

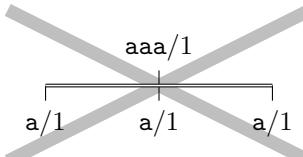
2.  $w = bbaa$  : on peut appliquer le dernier cas de la définition avec  $w_1 = b$ ,  $w_2 = b$  et  $w_3 = aa$ . Ceci est possible car  $\Psi_a(w_1) = \Psi_a(w_2) = \Psi_a(w_3) = 0$  est idempotent (**règle de l'idempotence**). Pour  $w_3$ , on utilise ensuite un arbre de factorisation similaire à celui vu plus haut pour  $u = ab$ .

On obtient alors l'arbre :



**Question 8.** Donner un arbre de factorisation pour  $w = \text{bbbaa}$  qui est différent de celui de l'exemple 6.

Considérons le mot **aaa**. Puisque,  $\Psi_a(a) = 1$  n'est pas idempotent dans le semi-groupe  $\mathbb{B}$ , il **n'est pas possible** d'appliquer la règle de l'idempotence. Autrement dit, l'arbre suivant n'est pas un arbre de factorisation de **aaa**.



cet arbre n'est pas valide

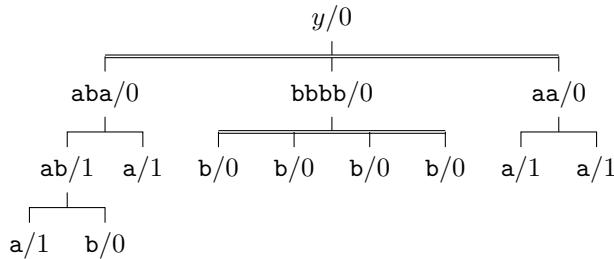
**Question 9.** Donner un arbre de factorisation de **aaa**.

**Question 10.** Donner un arbre de factorisation de **aaaaaa** (six « a ») qui utilise la règle de l'idempotence.

On associe à un arbre de factorisation sa **hauteur**, définie de la façon suivante :

- un arbre constitué uniquement d'une racine sans enfants est de hauteur 0,
- pour tous arbres  $\mathcal{A}_1, \dots, \mathcal{A}_k$  de hauteurs respectives  $h_1, \dots, h_k$ , l'arbre formé par une racine dont les enfants sont  $\mathcal{A}_1, \dots, \mathcal{A}_k$  est de hauteur  $1 + \max(h_1, \dots, h_k)$ .

**Exemple 7.** Poursuivons l'exemple 6. L'arbre de factorisation de **ab** est de hauteur 1, celui de **bbbaa** est de hauteur 2. Un exemple plus complexe est l'arbre de hauteur 3 donné ci-dessous pour  $y = \text{ababbbaa}$ .



On peut vérifier que pour tout morphisme de semi-groupe  $\phi : \Sigma^+ \rightarrow S$  et tout mot  $w \in \Sigma^+$ , on peut trouver un arbre de factorisation de hauteur au plus  $\lceil \log_2 |w| \rceil$  en coupant le mot en deux à chaque étage de l'arbre. Il n'est pas clair à priori qu'il soit possible de faire mieux en général. Un résultat surprenant et fondamental est que si  $S$  est **fini**, alors il est en fait toujours possible de trouver un arbre de factorisation de hauteur constante, c'est-à-dire qui ne dépend que de  $S$  mais pas de la longueur de  $w$ .

**Théorème 1.** Soit  $\Sigma$  un alphabet,  $(S, \cdot)$  un semi-groupe **fini** et  $\phi : \Sigma^+ \rightarrow S$  un morphisme de semi-groupe. Alors pour tout mot non vide  $w \in \Sigma^+$ , il existe un arbre de factorisation  $\mathcal{T}_w$  de  $w$  pour  $\phi$  de hauteur au plus  $3|S|$ . De plus, il existe un algorithme qui sur tout mot  $w \in \Sigma^+$  calcule  $\mathcal{T}_w$  en temps  $A|w|$ , où  $A$  est une constante qui ne dépend que de  $S$ .

### Partie III. Cas des groupes

Nous allons démontrer un cas particulier du Théorème 1. Pour tout entier  $k \geq 1$ , on identifie  $\mathbb{Z}/k\mathbb{Z}$  avec l'ensemble  $\{0, \dots, k-1\}$ . On rappelle que  $(\mathbb{Z}/k\mathbb{Z}, +)$  est un groupe où « + » est l'addition modulo  $k$ . L'inverse d'un élément  $x \in \mathbb{Z}/k\mathbb{Z}$  sera noté «  $-x$  ». On admettra que 0 est l'unique élément idempotent de  $(\mathbb{Z}/k\mathbb{Z}, +)$ . On remarque que l'addition modulo 2 se comporte comme la « OU » exclusif  $\oplus$ . On pourra ainsi identifier  $\mathbb{Z}/2\mathbb{Z}$  avec  $\mathbb{B}$  (voir page 2).

Dans cette partie,  $\Sigma$  est un alphabet fixé et  $k \geq 2$  un entier fixé. On pose  $S = (\mathbb{Z}/k\mathbb{Z}, +)$  et on se donne un morphisme de semi-groupe  $\phi : \Sigma^+ \rightarrow S$  quelconque. Pour chaque mot  $w \in \Sigma^+$ ,

$$P(w) = \{\phi(w[1, \dots, j]) : 1 \leq j \leq |w| - 1\}$$

est l'ensemble des valeurs par  $\phi$  des **préfixes stricts** de  $w$  (c'est-à-dire des préfixes non vides de  $w$  qui ne sont pas égaux à  $w$ ). On notera bien que  $\varepsilon$  et  $w$  ne sont pas des préfixes stricts de  $w$ .

**Exemple 8.** En identifiant  $\mathbb{Z}/2\mathbb{Z}$  et  $\mathbb{B}$ , la fonction  $\Psi_a$  définie à l'exemple 5 (page 2) peut être vue comme une fonction  $\Psi_a : \Sigma^* \rightarrow \mathbb{Z}/2\mathbb{Z}$ . Prenons  $w = \mathbf{bbaa}$ . Alors

$$P(w) = \{\Psi_a(\mathbf{b}), \Psi_a(\mathbf{bb}), \Psi_a(\mathbf{bba})\} = \{0, 0, 1\} = \{0, 1\}.$$

**Question 11.** On se place dans le contexte de l'exemple 8. Soit  $w = \mathbf{b}^{10}\mathbf{a}$  (dix «  $\mathbf{b}$  » suivis d'un «  $\mathbf{a}$  »). Donner  $P(w)$ . On justifiera la réponse.

On revient au cas général d'un morphisme de semi-groupe  $\phi : \Sigma^+ \rightarrow S$  où  $S = \mathbb{Z}/k\mathbb{Z}$  avec  $k \geq 2$ , et où  $\Sigma$  est un alphabet quelconque.

**Question 12.** Montrer que pour tous mots  $x, y \in \Sigma^+$ , on a  $\phi(y) = \phi(xy) - \phi(x)$ .

Pour  $p = 0, \dots, k$ , on note  $H(p)$  la proposition suivante :  
 « Pour tout mot  $w \in \Sigma^+$  tel que  $|P(w)| \leq p$ , il existe un arbre de factorisation de hauteur au plus  $3p$ . »

**Question 13.** Montrer que  $H(0)$  est vraie.

Prenons  $p \in \{1, \dots, k\}$  et supposons que  $H(p-1)$  est vraie. Soit  $t \in P(w)$  et notons  $j_1 < \dots < j_\ell$  les indices tels que  $\phi(w[1, \dots, j_i]) = t$  pour  $i = 1, \dots, \ell$ . On a ainsi  $\{j : \phi(w[1, \dots, j]) = t\} = \{j_1, \dots, \dots, j_\ell\}$ . On notera bien que  $\ell > 0$  puisque  $t \in P(w)$ . Écrivons alors

$$w = \underbrace{w[1, \dots, j_1]}_x \underbrace{w[j_1 + 1, \dots, j_2]}_{v_1} \dots \underbrace{w[j_{\ell-1} + 1, \dots, j_\ell]}_{v_{\ell-1}} \underbrace{w[j_\ell + 1, \dots, |w|]}_y.$$

Notons que dans l'écriture ci-dessus, il est possible que  $\ell = 1$  et donc que  $w = xy$ .

**Question 14.** Montrer que pour tout  $i = 1, \dots, \ell - 1$ , on a  $\phi(v_i) = 0$ .

**Question 15.** Montrer que  $|P(x)| < |P(w)|$ .

**Question 16.** Soit  $i = 1, \dots, \ell - 1$ . Montrer que  $|P(v_i)| < |P(w)|$ .

On admettra que  $|P(y)| < |P(w)|$  par un raisonnement similaire à celui de la question 16.

**Question 17.** Montrer que  $H(p)$  est vraie.

**Question 18.** En déduire que pour tout mot  $w \in \Sigma^+$ , il existe un arbre de factorisation de  $w$  pour  $\phi$  de hauteur au plus  $3|S|$ .

## Partie IV. Semi-groupe de Brandt

Nous nous intéressons à un autre cas particulier du Théorème 1. On fixe un entier naturel  $n \geq 1$  et on définit le **semi-groupe de Brant** ( $\mathcal{B}_n, \cdot$ ) par

$$\mathcal{B}_n = \{0_n\} \cup \{M_{i,j} : i, j \in \{1, \dots, n\}\}$$

où «  $\cdot$  » est la multiplication de matrices,  $0_n$  est la matrice identiquement zéro de taille  $n \times n$ , et  $M_{i,j}$  est la matrice de taille  $n \times n$  dont l'entrée  $(i, j)$  vaut 1, et dont toutes les autres entrées sont nulles (l'entrée  $(i, j)$  est ligne  $i$  et colonne  $j$ ). Par exemple

$$\mathcal{B}_2 = \left\{ \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \right\}.$$

On admettra l'identité suivante pour tous  $i, j, k, \ell \in \{1, \dots, n\}$  :

$$M_{i,j} \cdot M_{k,\ell} = \begin{cases} M_{i,\ell} & \text{si } j = k, \\ 0_n & \text{sinon.} \end{cases} \quad (2)$$

Ainsi  $\mathcal{B}_n$  est bien un semi-groupe.

**Question 19.** Montrer que  $x \in \mathcal{B}_n$  est idempotent si, et seulement si,  $x = 0$  ou  $x \in \{M_{i,i} : i = 1, \dots, n\}$ .

**Question 20.** Soient  $\Sigma = \{\mathbf{a}, \mathbf{b}\}$  et  $\phi : \Sigma^+ \rightarrow \mathcal{B}_2$  définie par  $\phi(\mathbf{a}) = M_{1,2}$  et  $\phi(\mathbf{b}) = M_{2,1}$ . Montrer que pour tout mot  $w \in \Sigma^+$ , on a  $\phi(w) = M_{1,1}$  si et seulement si  $w \in (\mathbf{ab})^+$ .

**Question 21.** Soient  $\Sigma$  et  $\phi$  définis comme à la question 20. Montrer que tout mot de  $(\mathbf{ab})^+$  admet un arbre de factorisation pour  $\phi$  de hauteur au plus 2.

**Question 22.** Soit  $n \geq 1$ . Montrer que pour tous  $A, B, x \in \mathcal{B}_n$ , on a  $AB, AxB \in \{0_n, M_{A,B}\}$  où  $M_{A,B} \in \mathcal{B}_n$  est une matrice qui ne dépend que de  $A$  et  $B$  (et pas de  $x$ ) dont on explicitera la valeur. Montrer que si de plus  $BA \neq 0_n$ , alors  $M_{A,B}$  est idempotent.

Prenons  $\phi : \Sigma^+ \rightarrow \mathcal{B}_n$  un morphisme de semi-groupe et un mot  $w \in \Sigma^+$  de longueur  $|w| \geq 2$ . On suppose que  $\phi(w) \neq 0_n$  et on décompose  $w$  de la façon suivante :

$$w = \mathbf{a} \mathbf{b} u_1 \mathbf{a} \mathbf{b} u_2 \dots \mathbf{a} \mathbf{b} u_{k+1} \quad (3)$$

où  $\mathbf{a}, \mathbf{b} \in \Sigma$  sont des lettres,  $k \geq 0$  et où les mots  $u_1, \dots, u_{k+1} \in \Sigma^*$  ne contiennent pas le facteur  $\mathbf{ab}$ . On prêtera attention au fait que les mots  $u_i$  peuvent être vides et qu'on peut avoir  $\mathbf{a}=\mathbf{b}$ .

**Exemple 9.** Soit  $\Sigma = \{\mathbf{a}, \mathbf{b}, \mathbf{c}\}$ . Le mot  $w = \mathbf{a} \mathbf{b} \mathbf{a} \mathbf{b} \mathbf{c} \mathbf{a} \mathbf{a} \mathbf{a} \mathbf{b} \mathbf{c}$  se décompose comme en (3) en écrivant

$$w = \underbrace{\mathbf{a} \mathbf{b}}_{\mathbf{a} \mathbf{b}} \underbrace{\varepsilon}_{u_1} \underbrace{\mathbf{a} \mathbf{b}}_{\mathbf{a} \mathbf{b}} \underbrace{\mathbf{c} \mathbf{a} \mathbf{a}}_{u_2} \underbrace{\mathbf{a} \mathbf{b}}_{\mathbf{a} \mathbf{b}} \underbrace{\mathbf{c}}_{u_3}.$$

On revient au cas général d'une décomposition comme en (3) d'un mot  $w \in \Sigma^+$  de longueur  $|w| \geq 2$  et tel que  $\phi(w) \neq 0_n$ .

**Question 23.** Soit  $1 \leq i \leq k$ . Montrer que  $\phi(\mathbf{b} u_i \mathbf{a})$  est un élément idempotent dont la valeur ne dépend pas de  $u_i$ .

Supposons que  $k > 0$  et que tous les  $u_i$  sont non vides. Soient  $\mathcal{A}_1, \dots, \mathcal{A}_{k+1}$  des arbres de factorisation de  $u_1, \dots, u_{k+1}$  respectivement.

**Question 24.** Donner un arbre de factorisation de  $w$  de hauteur au plus  $5 + \max(h_1, \dots, h_{k+1})$ , où  $h_i$  désigne la hauteur de  $\mathcal{A}_i$  pour  $1 \leq i \leq k+1$ . On justifiera qu'il s'agit bien d'un arbre de factorisation.

Les autres cas ( $k = 0$ , un ou plusieurs  $u_i$  sont vides) peuvent être traités de manière similaire. On admettra que l'on peut obtenir un arbre de factorisation de  $w$  de hauteur au plus

$$5 + \max(h_1, \dots, h_{k+1}) \quad \text{où } h_i = \begin{cases} \text{hauteur de } \mathcal{A}_i & \text{si } u_i \neq \varepsilon, \\ 0 & \text{sinon.} \end{cases}$$

Afin d'obtenir les arbres  $\mathcal{A}_1, \dots, \mathcal{A}_{k+1}$ , on voit que l'on peut procéder récursivement de la même manière puisque  $\phi(u_i) \neq 0_n$ , en tant que facteur de  $w$ . Cette récurrence est bien fondée car  $|u_i| < |w|$ .

**Question 25.** Montrer qu'il existe une constante  $C$  ne dépendant que de  $\Sigma$  et telle que tout mot  $w \in \Sigma^+$  avec  $\phi(w) \neq 0_n$ , admet un arbre de factorisation de hauteur au plus  $C$ .

## Partie V. Recherche infixe

Dans cette partie, on fixe un alphabet  $\Sigma$  et langage régulier  $L \subseteq \Sigma^+$ . On rappelle que l'on numérote les lettres à partir de l'indice 1 et que pour tout  $w \in \Sigma^+$ , et indices  $1 \leq i \leq j \leq |w|$ , on note  $w[i, \dots, j]$  le facteur  $w[i] \dots w[j]$ . Dans cette partie, on admettra le Théorème 1.

Considérons le problème suivant :

- Entrée : un mot  $w \in \Sigma^+$  et une paire d'indice  $(i, j)$  tels que  $1 \leq i \leq j \leq |w|$ .
- Sortie : « OUI » si le facteur  $w[i, \dots, j]$  appartient à  $L$ , « NON » sinon.

**Exemple 10.** Prenons  $\Sigma = \{\mathbf{a}, \mathbf{b}\}$  et le langage  $L = \{w \in \Sigma^+ : w \text{ contient un nombre pair de } \mathbf{a}\}$ . Considérons les cas suivants :

- entrée  $w = \mathbf{aababb}$  et  $(i, j) = (1, 2)$  : la sortie est OUI car  $w[1, \dots, 2] = \mathbf{aa} \in L$ ,
- entrée  $w = \mathbf{aababb}$  et  $(i, j) = (3, 5)$  : la sortie est NON car  $w[3, \dots, 5] = \mathbf{bab} \notin L$ ,
- entrée  $w = \mathbf{abba}$  et  $(i, j) = (1, 4)$  : la sortie est OUI car  $w[1, \dots, 4] = \mathbf{abba} \in L$ .

Le problème qui nous intéresse est le suivant, que l'on appellera **RECHERCHE-INFIXE<sub>L</sub>** :

- Entrée : un mot  $w \in \Sigma^+$  et une **liste** de paires d'indices  $I = [(i_1, j_1); \dots; (i_n, j_n)]$  tels que pour tout  $k \in \{1, \dots, n\}$ , on a  $1 \leq i_k \leq j_k \leq |w|$ .
- Sortie : une liste  $(b_1, \dots, b_n)$  où  $b_k = \text{OUI}$  si le facteur  $w[i_k, \dots, j_k]$  appartient à  $L$ , sinon  $b_k = \text{NON}$ .

**Exemple 11.** Prenons  $\Sigma = \{\mathbf{a}, \mathbf{b}\}$  et le langage  $L = \{w \in \Sigma^+ : w \text{ contient un nombre pair de } \mathbf{a}\}$ . On considère l'entrée donnée par le mot  $w = \mathbf{aababb}$  et la liste  $[(1, 2), (3, 5), (1, 4), (2, 6)]$  (de longueur  $n = 4$ ). La sortie attendue est donc (OUI, NON, NON, OUI) car :

- $w[1, \dots, 2] = \mathbf{aa}$ , qui appartient bien à  $L$ ,
- $w[3, \dots, 5] = \mathbf{bab}$ , qui n'appartient pas à  $L$ ,
- $w[1, \dots, 4] = \mathbf{aaba}$ , qui n'appartient pas à  $L$ ,
- $w[2, \dots, 6] = \mathbf{ababb}$ , qui appartient bien à  $L$ .

On cherche d'abord à donner une solution simple mais peu efficace à ce problème. Pour ce faire, prenons un AFD  $\mathcal{A} = (Q, q_0, \delta, F)$  qui reconnaît le langage  $L$ . Comme  $L$  est fixé,  $\mathcal{A}$  est aussi fixé. En particulier,  $|Q|$  est une constante du problème. On supposera dans la suite que l'on peut calculer  $\delta(q, \sigma)$  en temps constant pour tout  $q \in Q$  et toute lettre  $\sigma \in \Sigma$ .

Soit  $w \in \Sigma^*$  un mot. Le tableau  $T_w$  est défini pour tous  $1 \leq i \leq j \leq |w|$  et pour tout  $q \in Q$  par

$$T_w[q, i, j] = \delta^*(q, w[i, \dots, j]).$$

**Question 26.** Donner un algorithme qui calcule  $T_w$  en temps  $O(|Q| \cdot |w|^2)$ .

**Question 27.** Donner un algorithme qui résout toute instance  $(w, I)$  de **RECHERCHE-INFIXE<sub>L</sub>** en temps  $O(|Q| \cdot |w|^2 + n)$ , où  $n$  est la longueur de la liste  $I$ . On justifiera la correction de l'algorithme.

On cherche maintenant à donner une solution efficace à ce problème grâce aux forêts de factorisation.

Prenons le semi-groupe fini  $(S, \cdot)$  donné par le Lemme 1 (page 3) appliqué à  $L$ . Ainsi, il existe un ensemble  $F \subseteq S$  et un morphisme de semi-groupe  $\phi : \Sigma^+ \rightarrow S$  tel que  $\phi^{-1}(F) = L$ . Puisque  $S$  est fini et ne dépend que de  $L$  qui est fixé, on suppose dans la suite que l'on peut multiplier deux éléments de  $Q$  avec «  $\cdot$  » en temps constant. Par le Théorème 1, pour chaque mot  $w \in \Sigma^+$ , il existe un arbre de factorisation  $T_w$  de  $w$  pour  $\phi$ . De plus,  $T_w$  est de hauteur au plus  $3|S|$  et il existe un algorithme qui calcule  $T_w$  en temps  $O(|w|)$ .

**Dans toute la suite** on suppose que l'on dispose des fonctions **Racine**, **Indices**, **Enfants**, **Val** et **EnfantIndice** décrites ci-dessous.

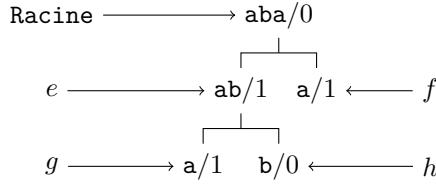
- **Racine**( $w$ ) renvoie, pour tout  $w \in \Sigma^+$ , un arbre de factorisation  $\mathcal{T}_w$  de  $w$  pour  $\phi$ .

Soit  $w \in \Sigma^+$  un mot et  $\mathcal{T}_w$  l'arbre de factorisation renvoyé par **Racine**( $w$ ). Si un nœud  $n$  de  $\mathcal{T}_w$  correspond au facteur  $w[i, \dots, j]$  de  $w$  et possède  $N$  enfants alors

- **Indices**( $n$ ) renvoie la paire  $(i, j)$ ,
- **Enfant**( $n, \ell$ ) renvoie, pour tout rang  $1 \leq \ell \leq N$ , le  $\ell$ -ième enfant de  $n$ ,
- **Val**( $n$ ) renvoie  $\phi(w[i, \dots, j])$ ,
- **EnfantIndice**( $n, k$ ) renvoie, pour tout indice  $1 \leq i \leq j$ , le rang  $\ell$  de l'unique enfant de  $n$  qui contient l'indice  $k$ , c'est-à-dire l'unique  $\ell$  tel que **Indices**(**Enfant**( $n, \ell$ )) =  $(i', j')$  avec  $i' \leq k \leq j'$ . Si  $n$  n'a aucun enfant alors cette fonction provoque une erreur.

On suppose que toutes ces fonction s'exécutent en temps constant.

**Exemple 12.** Avec le morphisme  $\Psi_a$  de l'exemple 5 (page 2), un arbre de factorisation du mot  $w = aba$  est le suivant, où l'on a annoté les nœuds de l'arbre avec des flèches pour leur donner des noms :



Faisons quelques remarques sur cet arbre :

- Puisque la racine a deux enfants  $e$  et  $f$ , on a **Enfant**(Racine, 1) =  $e$  et **Enfant**(Racine, 2) =  $f$ .
- La racine correspond au mot entier, c'est-à-dire à  $w = w[1, \dots, 3]$  donc **Indices**(Racine) =  $(1, 3)$ .
- La valeur de la racine est  $\Psi_a(aba) = 0$  donc **Val**(Racine) = 0.
- L'enfant gauche  $e$  correspond au facteur  $w[1, \dots, 2] = ab$  donc **Indices**( $e$ ) =  $(1, 2)$ .
- L'enfant droit  $f$  correspond au facteur  $w[3, \dots, 3] = a$  donc **Indices**( $f$ ) =  $(3, 3)$ .
- Enfin, **EnfantIndice**(Racine, 1) = **EnfantIndice**(Racine, 2) = 1 puisque le premier enfant de la racine ( $f$ ) correspond à la plage d'indices  $(1, 2)$ . D'autre part, **EnfantIndice**(Racine, 3) = 2 puisque le deuxième enfant correspond à la plage d'indices  $(3, 3)$ .

**Question 28.** Donner les valeurs de **Val**( $e$ ), **Val**( $f$ ), **Indices**( $g$ ), **Indices**( $h$ ), **EnfantIndice**( $e, 1$ ) et **EnfantIndice**( $e, 2$ ).

Soit  $w \in \Sigma^+$  un mot et  $\mathcal{T}_w$  l'arbre de factorisation renvoyé par **Racine**( $w$ ). La figure 1 page 11 présente l'algorithme **CalculePhi**( $n, i, j$ ) où  $n$  est un nœud de  $\mathcal{T}_w$  tel que **Indices**( $n$ ) =  $(I, J)$  avec  $I \leq i \leq j \leq J$ . On se propose de montrer que **CalculePhi**( $n, i, j$ ) renvoie  $\phi(w[i, \dots, j])$ . À cette fin, étant donné un nœud  $n$  de  $\mathcal{T}_w$ , on considère la proposition

$\text{HR}_w(n)$  : « Pour tous indices  $(i, j)$  tels que  $I \leq i \leq j \leq J$ , avec  $(I, J) = \text{Indices}(n)$  , on a  $\text{CalculePhi}(n, i, j) = \phi(w[i, \dots, j])$ . »

**Question 29.** Montrer que  $\text{HR}_w(n)$  est vraie pour tout mot  $w \in \Sigma^+$  et pour toute feuille  $n$  de  $\mathcal{T}_w$ .

Fixons un mot  $w$ . On montre  $\text{HR}_w(n)$  par induction sur la hauteur du nœud  $n$  dans l'arbre  $\mathcal{T}_w$ . Considérons un nœud  $n$  qui n'est pas une feuille et supposons que  $\text{HR}_w(f)$  est vraie pour tout enfant  $f$  de  $n$ . On cherche à montrer que  $\text{HR}_w(n)$  est vraie. Pour cela, prenons  $(i, j)$  tels que  $I \leq i \leq j \leq J$ , avec  $(I, J) = \text{Indices}(n)$ .

**Question 30.** Montrer que si l'algorithme atteint la ligne 13, alors  $v_f = \phi(w[i, \dots, p])$ .

---

**Fonction** CalculePhi( $n, i, j$ )

---

**Entrées** : : noeud  $n$ , indices  $i \leq j$   
**Sorties** : : la valuer de  $\phi(w[i, \dots, j])$

1 **début**

2     $(I, J) \leftarrow \text{Indices}(n)$

3    **si**  $i = I$  et  $j = J$  **alors**

4      $\leftarrow \text{renvoyer } \text{Val}(n)$

5     $\ell \leftarrow \text{Enfant Indice}(n, i)$

6     $r \leftarrow \text{Enfant Indice}(n, j)$

7    **si**  $\ell = r$  **alors**

8      $\leftarrow \text{// } w[i, \dots, j] \text{ est contenu dans le } \ell\text{-ième enfant de } n$

9      $\leftarrow \text{renvoyer } \text{CalculePhi}(\text{Enfant}(n, \ell), i, j)$

10     $f \leftarrow \text{Enfant}(n, \ell)$

11     $g \leftarrow \text{Enfant}(n, r)$

12     $(\_, p) \leftarrow \text{Indices}(f) \text{ //on ignore l'indice de gauche}$

13     $(q, \_) \leftarrow \text{Indices}(g) \text{ //on ignore l'indice de droite}$

14     $v_f \leftarrow \text{CalculePhi}(f, i, p)$

15     $v_g \leftarrow \text{CalculePhi}(g, q, j)$

16    **si**  $\ell = r$  **alors**

17      $\leftarrow \text{// } w[i, \dots, j] \text{ est contenu dans les } \ell\text{-ième et } (\ell + 1)\text{-ième enfants de } n$

18      $\leftarrow \text{renvoyer } v_f \cdot v_g$

17 **sinon**

18      $\leftarrow \text{// } w[i, \dots, j] \text{ est contenu dans les enfants des indices } \ell \text{ à } r$

18      $\leftarrow \text{renvoyer } v_f \cdot \text{Val}(\text{Enfant}(n, \ell + 1)) \cdot v_g$

---

FIGURE 1 – L'algorithme CalculePhi

On admettra qu'un raisonnement similaire permet de montrer que sous les mêmes hypothèses, on a  $v_d = \phi(w[q, \dots, j])$  à la ligne 14. Nous allons seulement nous intéresser à un cas possible, qui est le plus difficile. Il s'agit du cas où les conditions aux lignes 3, 7 et 15 ne sont pas satisfaites. Dans ce cas, l'algorithme va renvoyer une valeur à la ligne 18.

**Question 31.** Montrer que si l'algorithme atteint la ligne 18 alors il renvoie  $\phi(w[i, \dots, j])$ .

On admet que les autres cas de la preuve se traitent de façon similaire et que  $HR_w(n)$  est donc vraie.

**Question 32.** Montrer que pour tous  $1 \leq i \leq j \leq |w|$ , `CalculePhi(Racine(w), i, j)` s'exécute en temps  $O(2^h)$  où  $h$  est la hauteur  $T_w$ .

**Question 33.** En déduire qu'il existe une constante  $A$ , qui ne dépend que de  $L$ , telle que pour chaque  $w \in \Sigma^+$ , si l'arbre  $T_w$  est donné alors on peut calculer  $\phi(w[i, \dots, j])$  en temps  $A$  pour tous  $i$  et  $j$ .

**Question 34.** Donner un algorithme pour `RECHERCHE-INFIXE`  $L$ , qui résout toute instance  $(w, I)$  en temps  $O(|w| + |I|)$ .