Option Informatique en Spé MP et MP*

Devoir à rendre après les vacances de Noël

Morphismes, L-systèmes, lettres récurrentes

Résumé

▶ La notion de morphisme s'introduit naturellement, dans la structure algébrique A^*. Dans une première partie, on observe le lien avec le calcul matriciel classique.

▶ Un morphisme est itérable ; on s'intéresse dans une deuxième partie aux lettres *récurrentes* pour un morphisme, et on propose un algorithme de détermination de ces lettres, basé sur le calcul matriciel.

▶ Aristid Lindenmayer a été le premier à étudier le langage formé par les images d'une lettre (ou d'un mot) par les itérés d'un morphisme. On appelle désormais L-système un tel langage. La troisième partie étudie quelques questions simples sur les L-systèmes.

▶ Dans la quatrième partie, on s'intéresse au problème suivant : comment déterminer de manière efficace la i-ième lettre du mot $\varphi^n(a)$.

▶ Enfin, la cinquième partie propose une mise en œuvre en Caml.

$\boxed{\textit{Veuillez rédiger chaque partie sur une copie séparée.}}$

Table des matières

1 Morphismes

▶ Dans tout ce problème, A est un alphabet fini, contenant au moins les lettres a et b.

▶ Un *morphisme* est une application φ de A^* dans lui-même vérifiant $\varphi(vw) = \varphi(v)\varphi(w)$ quels que soient les mots u et v. On a donc $\varphi(\varepsilon) = \varepsilon$, et $\varphi(v^n) = \big(\varphi(v)\big)^n$ quels que soient $u \in \mathsf{A}^*$ et $n \in \mathbb{N}$. Il est clair que φ est parfaitement défini par les images des lettres de A. La *largeur* du morphisme φ est le naturel $d_\varphi = \max\limits_{x\in\mathsf{A}}\big|\varphi(x)\big|$.

Question 1 Dans cette question, $\mathsf{A} = \{a,b\}$. On note ψ le morphisme de FIBONACCI, défini par $\psi(a) = ab$ et $\psi(b) = a$. On note f_n le mot $\psi^n(a)$, où ψ^n désigne le n-ième itéré de ψ. Calculez f_n pour $n \in [\![2, 5]\!]$, puis établissez une relation simple entre f_n, f_{n+1} et f_{n+2}.

▶ Au morphisme φ, on associe la matrice M_φ, carrée d'ordre $|\mathsf{A}|$, définie par $(M_\varphi)_{\ell,k} = \big|\varphi(k)\big|_\ell$. Les lignes et les colonnes de M sont indexées par les lettres de A ; et l'élément de la ligne ℓ, colonne k, donne le nombre d'occurrences de la lettre ℓ dans le mot $\varphi(k)$. Il est clair que les coefficients de M_φ sont tous dans $\mathbb{N}$. On utilisera l'ordre naturel sur les lettres de A : la première ligne sera indexée par a, la deuxième par b et ainsi de suite.

Question 2 • Explicitez la matrice M_ψ associée au morphisme de FIBONACCI, défini à la question 1.

▶ Au mot $v \in \mathsf{A}^*$, on associe l'élément $\mathbf{V}$ de $\mathcal{M}_{|\mathsf{A}|,1}(\mathbb{N})$ défini par $\mathbf{V}_{\ell,1} = |v|_\ell$ pour tout $\ell \in \mathsf{A}$.

Question 3 • Justifiez la relation $\big|\varphi(v)\big|_x = \big(M_\varphi \cdot \mathbf{V}\big)_x$.

Question 4 • Conformément à l'usage, on identifie un élément $\mathbf{M}$ de $\mathcal{M}_{1,1}(\mathbb{N})$ et le scalaire $\mathbf{M}_{1,1}$. On note $\mathbf{H}$ l'élément de $\mathcal{M}_{1,|\mathsf{A}|}(\mathbb{N})$ défini par $\mathbf{H}_{1,k} = 1$ pour tout $k \in \mathsf{A}$. Justifiez la relation $\big|\varphi^n(v)\big| = \mathbf{H} \cdot (M_\varphi)^n \cdot \mathbf{V}$.

Question 5 Soit $v \in \mathsf{A}^*$. Justifiez l'affirmation suivante : on peut calculer $\big|\varphi^n(v)\big|$ en effectuant $\mathcal{O}(|\mathsf{A}|^3 \lg n)$ opérations élémentaires (additions et multiplications d'entiers naturels).

Question 6 • Soit $v \in \mathsf{A}^*$. Montrez que la suite de terme général $\ell_n = \big|\varphi^n(v)\big|$ vérifie une relation de récurrence linéaire d'ordre $|\mathsf{A}|$, c'est-à-dire qu'il existe une suite $(a_k)_{1\leqslant k\leqslant|\mathsf{A}|}$ de nombres vérifiant $\ell_n = \sum\limits_{1\leqslant k\leqslant|\mathsf{A}|} a_k \ell_{n-k}$ pour $n \geqslant |\mathsf{A}|$.

Question 7 • Prouvez que les a_k appartiennent tous à $\mathbb{Z}$.

2 Lettres récurrentes d'un morphisme

▶ Soient φ un morphisme et a, b deux lettres. On note $a \xrightarrow{\varphi} b$ s'il existe un exposant $n > 0$ tel que b possède au moins une occurrence dans le mot $\varphi^n(a)$.

Question 8 • Montrez que la relation $\xrightarrow{\varphi}$ est transitive.

▶ Soit u un mot sur l'alphabet A. On note $\alpha(u)$ l'ensemble des lettres qui possèdent au moins une occurrence dans u.

▶ Soient φ un morphisme et a une lettre. On définit une suite $(P_n)_{n\geqslant 1}$ de parties de A comme suit : $P_1 = \alpha\big(\varphi(a)\big)$, et $P_{n+1} = P_n \cup \alpha\big(\varphi^{n+1}(a)\big)$ pour $n \geqslant 1$. Ainsi, $x \in P_n$ ssi il existe un exposant $k \in [\![1, n]\!]$ tel que x possède au moins une occurrence dans le mot $\varphi^n(a)$.

Question 9 • Justifiez les affirmations suivantes : la suite $(P_n)_{n\geqslant 1}$ est croissante ; elle est stationnaire ; notant le rang n_0 à partir duquel elle stationne, on a $P_n \subsetneq P_{n+1}$ pour $1 \leqslant n < n_0$, et $n_0 \leqslant |\mathsf{A}|$.

Question 10 • Soient a et b deux lettres telles que $a \xrightarrow{\varphi} b$. Montrer qu'il existe un exposant n vérifiant $0 < n \leqslant |\mathsf{A}|$ et tel que b possède au moins une occurrence dans le mot $\varphi^n(a)$.

Question 11 • On note T_φ la matrice $\sum\limits_{1\leqslant k\leqslant|\mathsf{A}|} (M_\varphi)^k$. Montrez que $a \xrightarrow{\varphi} b$ ssi $(T_\varphi)_{b,a} > 0$.

▶ Une lettre a est *récurrente* pour φ si $a \xrightarrow{\varphi} a$.

Question 12 • Expliquez comment déduire de $T(\varphi, \mathsf{A})$ les lettres récurrentes pour φ. Quel est le coût de cette méthode, en nombre d'opérations élémentaires ?

Question 13 • Soit A une matrice carrée d'ordre p ; on note I_p la matrice unité d'ordre p et 0_p la matrice nulle d'ordre p. Donnez une expression simple de la puissance n-ième de la matrice ci-dessous, carrée d'ordre $2p$, décrite par blocs carrés d'ordre p :

$$\begin{pmatrix} \mathrm{I}_p & 0_p \\ A & A \end{pmatrix}$$

Question 14 • Déduisez de la question précédente un algorithme déterminant l'ensemble des lettres récurrentes pour φ, et ayant un coût $\mathcal{O}(|\mathsf{A}|^3 \lg |\mathsf{A}|)$.

3 L-systèmes

▶ Un *L-système* est un couple $\mathcal{G} = (\varphi, a)$ formé d'un morphisme et d'une lettre. Le *langage engendré* par $\mathcal{G}$ est l'ensemble $\{\varphi^n(a) \mid n \in \mathbb{N}\}$ des images de a par les itérés successifs de φ ; on le note $\mathcal{L}_{LS}(\mathcal{G})$ ou $\mathcal{L}_{LS}(\varphi, a)$.

Question 15 • Décrivez le langage $\mathcal{L}_{LS}(\varphi, a)$ lorsque φ est défini par $\varphi(a) = ab$ et $\varphi(b) = b$. Ce langage est-il rationnel ?

Question 16 • Décrivez le langage $\mathcal{L}_{LS}(\varphi, a)$ lorsque φ est défini par $\varphi(a) = ab$ et $\varphi(b) = bb$. Ce langage est-il rationnel ?

Question 17 • Définissez un morphisme φ tel que $\mathcal{L}_{LS}(\varphi, a) = \{a, ab, abc\}$.

Question 18 • Montrez que l'on peut, sans perte de généralité, se ramener au cas où $a \xrightarrow{\varphi} x$ pour toute lettre $x \in \mathsf{A}$. On supposera désormais que cette condition est vérifiée.

Question 19 • Soit (φ, a) un L-système. Montrez qu'une et une seule des deux assertions suivantes est vraie :

1. $\varphi^n(a) \neq \varepsilon$ quel que soit $n \in \mathbb{N}$;
2. il existe un rang $n_0 > 0$ tel que $\varphi^n(a) = \varepsilon$ pour tout $n > n_0$, et $\varphi^{n_0}(a) \neq \varepsilon$.

▶ Dans le premier cas de figure envisagé à la question précédente, le L-système $\mathcal{G} = (\varphi, a)$ est dit *immortel*. Dans le second cas, le L-système est dit *mortel* ; sa *durée de vie* est le naturel n_0.

Question 20 • Montrez que $\mathcal{G}$ est immortel ssi l'une au moins des lettres de l'alphabet utilisé est récurrente pour φ.

Question 21 • Quelle est la durée de vie maximale d'un L-système mortel ?

▶ Une lettre x est *fortement récurrente* pour φ s'il existe un exposant $n > 0$ tel que $\left|\varphi^n(x)\right|_x \geqslant 2$.

Question 22 • Montrez que s'il existe une lettre x fortement récurrente pour φ, alors le langage $\mathcal{L}_{LS}(\mathcal{G})$ est infini. La réciproque est-elle vraie ?

4 Vers l'algorithme de Swart

▶ Soit $\mathcal{G} = (\varphi, a)$ un L-système. Soient $n \in \mathbb{N}$, v un mot et $i \in [\![1, |\varphi^n(v)|]\!]$; on note $\boldsymbol{\lambda}(\varphi, n, v, i)$ la i-ième lettre de $\varphi^n(v)$. On se propose de déterminer efficacement $\boldsymbol{\lambda}(\varphi, n, a, i)$. Il est clair que l'on peut supposer $|\varphi(a)| \geqslant 2$ et $n \geqslant 1$.

Question 23 Que pensez-vous de la méthode consistant à construire le mot $\varphi^n(a)$?

▶ Notons $v = \varphi(a)$, $p = |v|$ et $v = v_1 v_2 \dots v_p$.

Question 24 • On note $x_0 = \varepsilon$ et $x_k = \varphi^{n-1}(v_1 v_2 \dots v_k)$ pour $1 \leqslant k \leqslant p$. Justifiez l'existence d'un indice $k \in [\![1, p]\!]$ tel que $|x_{k-1}| < i \leqslant |x_k|$. Un tel indice est-il unique ?

Question 25 • Justifiez la relation $\boldsymbol{\lambda}(\varphi, n, a, i) = \boldsymbol{\lambda}(\varphi, n-1, v_k, i - |x_{k-1}|)$.

Question 26 • Décrivez alors un algorithme de calcul de $\boldsymbol{\lambda}(\varphi, n, a, i)$.

▶ Notons $\mathsf{Nops}(\varphi, n, x, i)$ le coût (exprimé en opérations élémentaires) du calcul de $\boldsymbol{\lambda}(\varphi, n, x, i)$ au moyen de cet algorithme. On se propose de majorer la valeur maximale $\mathbf{C}(\varphi, n)$ de ce coût :

$$\mathbf{C}(\varphi, n) = \max_{x \in \mathsf{A}} \left(\max_{1 \leqslant i \leqslant |\varphi^n(x)|} \mathsf{Nops}(\varphi, n, x, i) \right)$$

Nous donnerons un majorant de $\mathbf{C}(\varphi, n)$ en fonction de n, de $|\mathsf{A}|$ et de d_φ.

Question 27 • Justifiez la relation $\mathbf{C}(\varphi, n) \leqslant \mathbf{C}(\varphi, n-1) + \mathcal{O}\big(|\mathsf{A}|^3 d_\varphi \lg n\big)$.

Question 28 En déduire la majoration $\mathbf{C}(\varphi, n) \leqslant \mathcal{O}\big(n|\mathsf{A}|^3 d_\varphi \lg n\big)$.

Question 29 Expliquez brièvement comment améliorer l'algorithme de manière à réaliser la majoration $\mathbf{C}(\varphi, n) \leqslant \mathcal{O}\big(n|\mathsf{A}|^3 \lg d_\varphi \lg n\big)$.

5 Programmation en Caml

▶ Nous allons mettre en œuvre certains des algorithmes vus dans les parties précédentes. Vous trouverez en annexe une liste de fonctions de la bibliothèque Caml, qui peuvent vous simplifier la tâche. À la fin de chaque question, on donne un objectif de coût, exprimé en nombre (maximal) de lignes. Bien entendu, ces lignes doivent être d'une largeur raisonnable ...

▶ Un morphisme sera représenté par une `(char * string) list` ; par exemple, le morphisme φ de Fibonacci sera décrit par `let psi = [(`a`,"ab");(`b`,"a")];;`.

Question 30 • Rédigez en Caml une fonction de signature :

```
list_of_string : string -> char list
```

spécifiée comme suit : `list_of_string s` construit la liste des caractères qui composent la chaîne s. Par exemple, `list_of_string "abac"` rendra la liste `[`a`;`b`;`a`;`c`]`. Objectif : 2.

Question 31 • Rédigez en Caml une fonction de signature :

```
applique_morphisme : (char * string) list -> string -> string
```

spécifiée comme suit : `applique_morphisme phi u` calcule $\varphi(u)$. Objectif : 2.

Question 32 • Rédigez en Caml une fonction de signature :

```
itere_morphisme : (char * string) list -> string -> int -> string
```

spécifiée comme suit : `itere_morphisme phi u n` calcule $\varphi^n(u)$. Objectif : 3.

Question 33 • Rédigez en Caml une fonction de signature :

```
lambda : (char * string) list -> int -> string -> int -> char
```

spécifiée comme suit : `lambda phi n u i` calcule $\boldsymbol{\lambda}(\varphi, n, u, i)$ en appliquant la méthode «naïve» évoquée à la question 23. Objectif : 2.

Question 34 • Rédigez en Caml une fonction de signature :

```
produit_matrices : int vect vect -> int vect vect -> int vect vect
```

spécifiée comme suit : `produit_matrices r s` calcule le produit rs des matrices r et s. Objectif : 10.

Question 35 • Rédigez en Caml une fonction de signature :

```
exponentielle_matrice : int vect vect -> int int vect vect
```

spécifiée comme suit : `exponentielle_matrice r n` calcule r^n, où r est une matrice carrée. Vous ferez appel à l'algorithme d'exponentiation rapide utilisé à la question 5. Objectif : 15.

Question 36 • Rédigez en Caml une fonction de signature :

```
matrice_of_morphisme : (char * string) list -> int int vect vect
```

spécifiée comme suit : `matrice_of_morphisme phi` calcule M_φ. Objectif : 9.

Question 37 • Rédigez en Caml une fonction de signature :

```
lettres_récurrentes : (char * string) list -> char list
```

spécifiée comme suit : `lettres_récurrentes phi` dresse la liste des lettres récurrentes de φ.

Question 38 • Rédigez en Caml une fonction de signature :

```
lambda : (char * string) list -> int -> string -> int -> char
```

spécifiée comme suit : `lambda phi n u i` calcule la valeur de $\boldsymbol{\lambda}(\varphi, n, u, i)$ en appliquant l'une des méthodes évoquées aux questions 28 et 29.

FIN