Option Informatique en Spé MP et MP*
Devoir à rendre après les vacances de la Toussaint

1 L-systèmes : une rapide introduction

▶ Un L-*schéma* est un couple (Σ, φ) où Σ désigne un alphabet fini non vide et φ est un morphisme de Σ^* sur lui-même. Un L-*système* est un triplet $(\Sigma, \varphi, \omega)$ où (Σ, φ) est un L-schéma et ω un mot non vide sur Σ. Nous dirons que ω est l'*axiome* du L-système ; la *suite engendrée* est la suite $\left(\varphi^k(\omega)\right)_{k\in\mathbb{N}}$ des images de ω par les itérés de φ. Le *langage engendré* est l'image de cette suite.

▶ Donnons un exemple : $\Sigma = \{a\}$; φ est défini par $\varphi(a) = a^2$. Alors $\varphi^k(a) = a^{2^k}$ pour tout $k \in \mathbb{N}$.

▶ Dans tout le problème, les morphsimes considérés seront *non-effaçants*, c'est-à-dire vérifient $\varphi(x) \neq \varepsilon$ pour toute lettre x.

▶ Dans les deux questions suivantes, nous prenons $\Sigma = \{a,b\}$ et $\omega = a$; φ est défini par $\varphi(a) = b$ et $\varphi(b) = ab$. Les six premiers termes de la suite engendrée sont a, b, ab, bab, $abbab$ et $bababbab$.

Question 1 Établissez la relation $u_{k+2} = u_k u_{k+1}$, où $u_k = \varphi^k(\omega)$.

Question 2 Que pensez-vous de la suite de terme général $|u_k|$?

▶ Dans les deux questions suivantes, nous prenons $\Sigma = \{a,b,c\}$ et $\omega = a$; φ est défini par $\varphi(a) = abc^2$, $\varphi(b) = bc^2$ et $\varphi(c) = c$. Nous notons encore $u_k = \varphi^k(\omega)$.

Question 3 Calculez u_2, u_3 et u_4.

Question 4 Donnez une expression simple de $|u_k|$.

Question 5 Proposez un L-système $(\Sigma, \varphi, \omega)$ tel que, avec les notations précédentes, $|u_k| = (k+1)^3$.

2 Récurrence dans un L-système

▶ Commencez par lire le document intitulé ≪*Plus long sous-mot commun : ce qu'il faut savoir*≫.

▶ Soient $\mathcal{S} = (\Sigma, \varphi)$ un L-schéma et $n \in \mathbb{N}^*$. Nous dirons que le mot u (non vide) est *n-récurrent* pour $\mathcal{S}$ si $u \sqsubseteq \varphi^n(u)$; n est un *exposant de récurrence* de u. Un mot est *récurrent* s'il est n-récurrent pour au moins un $n > 0$. Nous noterons $R_n(\mathcal{S})$ l'ensemble des mots n-récurrents pour $\mathcal{S}$, et $R(\mathcal{S})$ l'ensemble des mots récurrents pour $\mathcal{S}$.

Question 6 Montrez que $R_n(\mathcal{S})$ est stable par concaténation.

Question 7 Deux mots u et v vérifient $uv \in R_n(\mathcal{S})$. Montrez que l'un au moins de ces deux mots appartient à $R_n(\mathcal{S})$.

▶ Un mot u (non vide) est *vivant* pour $\mathcal{S}$ si $\varphi^k(u) \neq \varepsilon$ pour tout k.

▶ Nous admettrons le résultat suivant, dû à Higman : si L est un langage infini sur un alphabet fini Σ, alors il existe deux mots u et v de L tels que $u \sqsubseteq v$.

Question 8 Montrez qu'un L-schéma admet au moins un mot récurrent ssi il existe au moins un mot vivant.

Question 9 Montrez que si le mot u est récurrent, alors l'une au moins de ses lettres est récurrente.

▶ Un mot u est *n-récurrent minimal* s'il est n-récurrent, et s'il est de longueur minimale parmi les mots n-récurrents.

Question 10 Quels sont les mots n-récurrents minimaux?

▶ Nous dirons que le L-schéma $\mathcal{S}$ est récurrent si tout mot est récurrent pour $\mathcal{S}$.

Question 11 Soit $\mathcal{S}$ un L-schéma récurrent quelconque. Prouvez l'existence d'un $n > 0$ tel que tout mot soit n-récurrent pour $\mathcal{S}$.

Question 12 Dans cette question uniquement, $\mathcal{S}$ est le L-schéma (Σ, φ) défini par $\Sigma = \{a,b,c,d\}$, $\varphi(a) = bd$, $\varphi(b) = c$, $\varphi(c) = a$ et $\varphi(d) = a$. Déterminez le plus petit n tel que tout mot soit n-récurrent pour $\mathcal{S}$.

▶ Soient $\mathcal{S} = (\Sigma, \varphi)$ un L-schéma récurrent quelconque. Pour toute lettre x, nous noterons n_x le plus petit exposant n tel que x soit n-récurrente.

Question 13 Montrez que $n_x \leqslant |\Sigma|$, et que cette majoration est optimale.

▶ Soit u un mot récurrent pour le L-schéma $\mathcal{S} = (\Sigma, \varphi)$. Nous noterons $\boldsymbol{E}(u)$ l'ensemble des exposants de récurrence de u, c'est-à-dire $\{n > 0 \mid u \sqsubseteq \varphi^n(u)\}$.

Question 14 Montrez que $\boldsymbol{E}(u)$ est stable pour l'addition ; et que si $p \in \boldsymbol{E}(u)$ et $q > 0$, alors $pq \in \boldsymbol{E}(u)$.

Question 15 Soient u et v deux mots (non vides). Montrez que $\boldsymbol{E}(u) \cap \boldsymbol{E}(v) \subset \boldsymbol{E}(uv) \subset \boldsymbol{E}(u) \cup \boldsymbol{E}(v)$.

Question 16 Montrez que $\displaystyle\bigcap_{|u|\geqslant 1} \boldsymbol{E}(u) = \bigcap_{x\in\Sigma} \boldsymbol{E}(x)$.

▶ Notons $\boldsymbol{E}(\mathcal{S})$ l'ensemble des $n > 0$ tels qu'ils existe au moins un mot u n-récurrent.

Question 17 Montrez que $\displaystyle\boldsymbol{E}(\mathcal{S}) = \bigcup_{x\in\Sigma} \boldsymbol{E}(x)$.

3 Un peu de programmation pour conclure

Question 18 Rédigez en Caml une fonction :

```
est_sous_mot : string -> string -> bool
```

spécifiée comme suit : `est_sous_mot u v` dit si u est un sous-mot de v. Vous justifierez en toute rigueur la validité de cette fonction.

Question 19 Rédigez en Caml une fonction :

```
flat_string_list : string list -> string
```

spécifiée comme suit : `flat_string_list l` concatène les éléments de la liste de chaînes ℓ, pour former une seule (longue) chaîne. Par exemple, `flat_string_list ["ab";"b";"cab"]` construit la chaîne `"abbcab"`.

▶ Un morphisme φ sera représenté par une liste de couples du type `char * string`. Par exemple, le morphisme φ étudié aux questions 1 et 2 sera décrit par la liste `[(‘a‘, "b");(‘b‘,"ab")]`.

Question 20 Rédigez en Caml une fonction :

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

spécifiée comme suit : `calcule_image phi s` calcule l'image, par le morphisme décrit par la liste φ, du mot décrit par la chaîne s.

Question 21 Rédigez en Caml une fonction :

```
expo_rec : char -> (char * string) list -> int option
```

spécifiée comme suit : si la lettre a n'est pas récurrente, `expo_rec a` rend la valeur `None` ; sinon, `expo_rec a` rend la valeur `Some n` où n est le plus petit exposant de récurrence de a.

FIN