

**REQUIN**  
**recueil de questions d'informatique**

**Sommaire**

Chapitre I. Algorithmique .....	3
Chapitre II. Arbres & Graphes .....	5
Chapitre III. Langages formels .....	7
Chapitre IV. Théorie des jeux .....	9
Chapitre V. Calculabilité .....	11
Chapitre VI. Logique .....	13
Chapitre VII. Langages fonctionnels .....	15
Chapitre VIII. Mathématiques pour l'informatique .....	17
Problème VIII.1: Monoïdes libres, langages et actions .....	18

# Chapitre I. Algorithmique

*c'est si vide ici...*

## Chapitre II. Arbres & Graphes

*c'est si vide ici...*

## Chapitre III. Langages formels

*c'est si vide ici...*



## Chapitre IV. Théorie des jeux

*c'est si vide ici...*

# Chapitre V. Calculabilité

*c'est si vide ici...*

# Chapitre VI. Logique

*c'est si vide ici...*

# Chapitre VII. Langages fonctionnels

*c'est si vide ici...*



# Chapitre VIII. Mathématiques pour l'informatique

## Problème VIII.1: Monoïdes libres, langages et actions

### VIII.1.a Demi-groupes, monoïdes et groupes

Soit un *demi-groupe*  $(\mathbb{E}, +)$ , c'est-à-dire que

- $\mathbb{E}$  est stable par  $+$
- La loi  $+$  est associative

On dira de plus que  $\mathbb{E}$  est un *monoïde* si il existe  $e \in \mathbb{E}$  tel que

$$\forall x \in \mathbb{E}, xe = ex = x$$

On dira enfin que  $\mathbb{E}$  est un *groupe* si il existe  $\cdot^{-1} : \mathbb{E} \rightarrow \mathbb{E}$  tel que

$$\forall x \in \mathbb{E}, xx^{-1} = x^{-1}x = e$$

0. Donner un groupe, puis un monoïde qui n'est pas un groupe, et enfin un demi-groupe qui n'est pas un monoïde. ★★★★★

Si  $\mathbb{E}$  est un monoïde, soit  $\sim \in (\mathbb{E}^2)^2$  telle que  $(a, b) \sim (c, d) \iff a + d = b + c$ .

1. Que dire de  $\mathbb{E}^2 / \sim$  ? ★★★★★

Soit  $\Sigma$  un ensemble fini. On appelle  $\Sigma^*$  le plus petit monoïde contenant  $\Sigma$  et tel que

$$\forall u, v, w, x \in \Sigma^*, (u, v) = (w, x) \iff uv = wx$$

On note son neutre  $\varepsilon$ .

2. Justifier que  $\Sigma^*$  est l'ensemble des mots finis sur  $\Sigma$  ★★★★★

On pose  $\mathcal{A} := \{x \mapsto xw, w \in \Sigma^*\}$ , que l'on munit de la loi de composition usuelle des fonctions.

3. Justifier que  $\Sigma^*$  et  $\mathcal{A}$  sont isomorphes comme monoïdes. ★★★★★

### VIII.1.b Associativité ?

Dans cette partie,  $(S, +)$  est un demi-groupe.

Soit  $n \in \mathbb{N}$  puis  $a \in S^n$  un  $n$ -uplet.

4. Donner le langage des expressions calculant  $\sum a$ . Est-il rationnel ? ★★★★★

Par exemple, pour  $n = 3$ ,  $\mathcal{L} = \{a_1 + (a_2 + a_3), (a_1 + a_2) + a_3\}$ .

5. Mettre en bijection  $\mathcal{L}$  et l'ensemble des arbres binaires à  $n$  noeuds. Dénombrer  $\mathcal{L}$ . ★★★★★

On considère maintenant posséder une machine capable d'exécuter  $\omega \in \mathbb{N}^*$  opérations “+” simultanées.

6. Donner un mauvais ordre de calcul de  $\sum a$ , puis un choix plus raisonnable. ★★★★★

### VIII.1.c Retouches

Soient  $\mathcal{L}$  un langage rationnel et  $M \in \Sigma^*$  un mot de longueur  $n \in \mathbb{N}^*$ . On appelle une *requête* un couple  $1 \leq i \leq j \leq n$  et sa *taille* est  $r := j - i$ . On *satisfait* une requête en renvoyant si  $M[i : j] \in \mathcal{L}$ . On note  $q \in \mathbb{N}$  le nombre d'états d'un automate qui reconnaît  $\mathcal{L}$ .

7. Donner un algorithme satisfaisant une requête.

★★★★★

Moyennant un précalcul,

8. Donnez un algorithme efficace satisfaisant une requête en temps  $\mathcal{O}(q \log r)$

★★★★★

On pourra introduire un ensemble de fonctions similaire à  $\mathcal{A}$  agissant sur l'automate

Une *modification* est une opération de la forme  $M[i] \leftarrow a$  avec  $a \in \Sigma$ .

9. Modifier l'algorithme précédent pour permettre des modifications en temps  $\mathcal{O}(q \log r)$ .

★★★★★