Examen final de Calculabilité et Complexité

Consignes – Ce devoir doit être réalisé individuellement. Les exercices sont indépendants.

Exercice 1. Machines de Turing qui reconnaîssent des langages

Proposer des machine de Turing reconnaissant chacun des langages suivants.
a - $L_1 := \left\{w \in \{\mathtt{a},\mathtt{b}\}^\star \;\middle|\; \exists u, v \in \{\mathtt{a},\mathtt{b}\}^\star :\; w = u\,\mathtt{abba}\,v\right\}$
(les mots qui contiennent le motif abba).
b - $L_2 := \left\{w \in \{\mathtt{a},\mathtt{b}\}^\star \;\middle|\; |w|_{\mathtt{a}} = 2 \times |w|_{\mathtt{b}}\right\}$
(les mots qui ont deux fois plus de a que de b).
c - $L_3 := \left\{\mathtt{a}^{2^n} \;\middle|\; n \in \mathbb{N}\right\}$
(les séquences de a dont la longueur est une puissance de 2).

Exercice 2. Les carrés

Les machines de Turing peuvent reconnaître plus que les langages hors-contexte. C'est le cas par exemple du langage $L := \left\{ww \;\middle|\; w \in \{\mathtt{a},\mathtt{b}\}^\star\right\}$ des carrés. On peut montrer, par exemple avec le lemme de la double étoile, que ce langage n'est pas hors-contexte : il ne peut être défini à l'aide d'une grammaire hors-contexte.

Question 1. Écrire une machine de Turing reconnaissant le langage $w\mathtt{X}w$, où $w \in \{\mathtt{a},\mathtt{b}\}^\star$.

Question 2. Écrire une machine de Turing qui rejette les mots de longueur impaire, et pour les mots de longueur paire insère le symbole X entre les deux moitiés du mot. Par exemple, le mot abaa est transformé en abXaa.

Question 3. Combiner les machines des deux questions précédentes pour en produire une reconnaissant L.

Question 4. (Bonus) Montrer que le langage L n'est pas rationnel.

Exercice 3. Machines de Turing qui calculent

On va manipuler des listes d'entiers, encodées par la fonction $\langle - \rangle_{code}$ comme suit :

- les entiers sont encodés en binaire avec le bit de poids fort en tête ;
- les éléments d'une liste sont séparés par le symbole \$;
- la liste est délimitée à gauche et à droite par un symbole \$.

Par exemple :

$$\langle [] \rangle_{code} := \$\$ \qquad \langle [3] \rangle_{code} := \$11\$$$
$$\langle [3,2] \rangle_{code} := \$11\$10\$ \qquad \langle [1,2,3,4,5,6] \rangle_{code} := \$1\$10\$11\$100\$101\$110\$$$

Question 1. Proposer une machine de Turing qui calcule la fonction `rev`, qui inverse l'ordre des éléments d'une liste (d'entiers). Par exemple, comme $\mathtt{rev}[1,2,3] = [3,2,1]$, on veut une machine qui, si elle est exécutée sur $\langle [1,2,3] \rangle_{code}$, produit $\langle [3,2,1] \rangle_{code}$.

Question 2. Proposer une machine de Turing qui calcule la somme des entiers contenus dans une liste. Par exemple, sur l'entrée $\langle[1, 2, 3]\rangle_{code}$, la machine devra produire le mot `110`, qui encode l'entier $1 + 2 + 3 = 6$.

Question 3. Proposer une machine de Turing qui calcule la longueur d'une liste (d'entiers). Par exemple, sur l'entrée $\langle[1, 2, 3]\rangle_{code}$, elle produira `11`, c'est à dire la représentation de $3 = |[1, 2, 3]|$.

Exercice 4. Difficulté des problèmes

Pour chacun des problèmes suivants, indiquer s'ils sont :

1. rationnels
2. non-rationnels mais décidables
3. indécidables mais reconnaissables
4. ni indécidables, ni reconnaissables.

Il est nécessaire dans cet exercice de justifier soigneusement vos réponses. On appréciera tout particulièrement les justifications *claires* et *concises* (également : *justes*).

Dans la suite, on fixe $\Sigma = \{\mathtt{a}, \mathtt{b}\}$, et on note $\mathcal{MT}(\Sigma)$ l'ensemble des machines de Turing sur l'alphabet Σ.

a - Machines reconnaissant `ab` :

Instances : $\mathcal{M} \in \mathcal{MT}(\Sigma)$

Instances Positives : $\mathcal{M}$ telle que $\mathtt{ab} \in \mathcal{L}(\mathcal{M})$.

b - Machines reconnaissant au moins un mot de taille bornée :

Instances : paires $\langle\mathcal{M}, n\rangle$, avec $\mathcal{M} \in \mathcal{MT}(\Sigma)$ et $n \in \mathbb{N}$

Instances Positives : $\langle\mathcal{M}, n\rangle$ tels que il existe $w \in \Sigma^\star$ tel que $|w| < n$ et $w \in \mathcal{L}(\mathcal{M})$.

c - Paires de machines ayant au moins un mot en commun :

Instances : paires $\langle\mathcal{M}_1, \mathcal{M}_2\rangle \in \mathcal{MT}(\Sigma) \times \mathcal{MT}(\Sigma)$,

Instances positives : paires $\langle\mathcal{M}_1, \mathcal{M}_2\rangle$ telles que $\mathcal{L}(\mathcal{M}_1) \cap \mathcal{L}(\mathcal{M}_2) = \emptyset$.

d - Machines avec plus d'états finaux que d'états non-finaux :

Instances : $\mathcal{M} \in \mathcal{MT}(\Sigma)$,

Instances positives : $\mathcal{M} = \langle Q, \Sigma, \Delta, q_0, F\rangle$ telle que $|F| \geqslant |Q \setminus F|$.

e - Machines sans états finaux :

Instances : $\mathcal{M} \in \mathcal{MT}(\Sigma)$,

Instances positives : $\mathcal{M} = \langle Q, \Sigma, \Delta, q_0, F\rangle$ telle que $F = \emptyset$.