TD1 : Machines de Turing

1 Core Machines

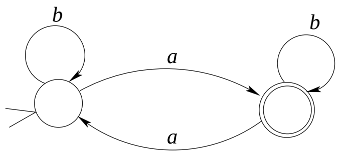
Dans les exercices de cette section, vous définirez des machines de Turing à partir de la spécification donnée en n'utilisant que les définitions de base et sans faire de composition de machines.

1- Construisez une machine de Turing $M_t = (K, \Sigma, \delta, s, \{y, n\})$ qui avec la tête dans une position donnée cherche vers la droite s'il y a le caractère $ sur le ruban. La machine s'arrête dans l'état y si le caractère $ est trouvé, et dans l'état n si la tête de lecture a lu deux blancs consécutifs sur le ruban.

2- On reprend l'idée de l'exercice précédent, mais cette fois-ci, on veut une machine de Turing qui répond à la spécification suivante :
- le caractère à chercher est celui qui est sous la tête de lecture au démarrage de la machine (si c'est un blanc, la machine s'arrête tout de suite dans l'état n) ;
- la machine cherche donc une deuxième occurence de ce caractère sur le ruban, cette recherche se fait aussi bien dans la partie droite que dans la partie gauche du ruban ;
- la recherche s'arrête lorsque la machine trouve une seconde occurence du caractère cherché ou si elle lit deux blancs qui ne sont pas forcément consécutifs et qui sont situés *à droite* de la position de départ de la machine.

3- Construisez une machine de Turing qui part d'une position, change les a en b et les b en a en allant vers la droite jusqu'à rencontrer un blanc, puis replace la tête de lecture à la position initiale.

4- On considère l'automate défini par le schéma suivant :



Construisez une machine de Turing qui “mime” le fonctionnement de cet automate.

Généralisez votre réponse en indiquant un moyen de transformer tout automate déterministe fini en une machine de Turing. Qu'en est-il des automates indéterministes ? des automates à pile ?

2 Double action

On considère une variante des machines de Turing, que nous allons appeler machine de Turing à double action (MT2), pour lesquelles la tête de lecture est systématiquement déplacée dans un sens de *parcours courant* (vers la droite ou vers la gauche).

a) Précisez quelles sont les actions possibles pour une telle machine.

b) Donnez une définition précise pour les machines MT2.

c) Montrez que les machines de Turing et les machines MT2 sont équivalentes dans un sens que vous préciserez.

3 Machines de Turing et calculs

1- Soit $\Sigma = \{a, b, c\}$, construisez une machine de Turing calculant la fonction $f : \Sigma^* \to \Sigma^*$ qui à tout mot w associe le mot w' qui est le "projeté" de w sur $\{b, c\}$ (c'est-à-dire le mot issu de w après gommage de la lettre a).

2- Soit $\Sigma = \{a, b, c\}$, construisez une machine de Turing calculant la fonction $f : \Sigma^* \to \Sigma^*$ qui à tout mot w associe le mot ww^R.

3- Construisez une machine de Turing qui calcule la fonction $f : \mathbb{N} \to \mathbb{N}$ telle que $f(x) = x - 1$ si $x \neq 0$ et $f(0) = 0$.

4- Construisez une machine de Turing qui décide le langage L dans les cas suivants :
- $L = a^*ba^*b$
- $L = \{w \in \{a, b, c\}^* | w \text{ contient } abc \text{ mais pas } bb\}$
- $L = \{a^n b^n | n \in \mathbb{N}\}$
- $L = \{ww | w \in \{a, b\}^*\}$

4 Composition de machines de Turing

1- Construisez une machine de Turing sous forme de combinaison de machines de Turing élémentaires qui fasse la soustraction de deux entiers écrits en notation unaire. Plus précisément, cette machine calculera la fonction $f : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ telle que $f(n, m) = n - m$ si $m \leq n$ et 0 sinon.

2- Même chose pour la multiplication (en unaire).

3- Écrivez une machine de Turing qui réalise la conversion unaire $\to$ binaire.

4- Soit Σ un alphabet, on appelle homomorphisme de monoïdes toute application φ de Σ^* dans Σ^* telle que $\varphi(uv) = \varphi(u)\varphi(v)$.

a) Est-ce que la fonction de la question 1.1 est un homorphisme de monoïdes ?
b) Montrez que tout homormorphisme de monoïdes est défini par l'image des lettres de l'alphabet. Donnez des exemples d'homomorphismes de monoïdes.
c) On considère l'alphabet $\Sigma = \{a, b, c\}$ et l'homomorphisme φ tel que $\varphi(a) = e$, $\varphi(b) = ab$ et $\varphi(c) = c$. Construisez une machine de Turing calculant φ.