Le sujet peut être traité dans n'importe quel ordre mais il faut indiquer devant chaque réponse la question correspondante. Les questions sont approximativement par difficulté croissante sauf celles notées ★ qui sont plus difficiles.

1 Lemmes de pompage

Soit $\mathbb{N} := \{0, 1, \ldots\}$. Pour $(a, b, c) \in \mathbb{N}^3$ on définit $L_{a,b,c} := \{0^{an}1^{bn}2^{cn} : n \geq 0\}$ sur l'alphabet $\Sigma := \{0, 1, 2\}$.

1.1. Trouver tous les triplets $(a, b, c) \in \mathbb{N}^3$ tels que $L_{a,b,c}$ est régulier.

  1.1.1) Pour les $L_{a,b,c}$ qui sont réguliers, justifier en donnant des automates ou des expressions régulières.

  1.1.2) Pour les $L_{a,b,c}$ qui ne sont pas réguliers, justifier en utilisant le lemme de pompage.

1.2. Trouver tous les triplets $(a, b, c) \in \mathbb{N}^3$ tels que $L_{a,b,c}$ est algébrique.

  1.2.1) Pour les $L_{a,b,c}$ qui ne sont pas algébriques, donner des grammaires ou des automates.

  1.2.2) Pour les $L_{a,b,c}$ qui ne sont pas algébriques, justifier en utilisant le lemme de pompage.

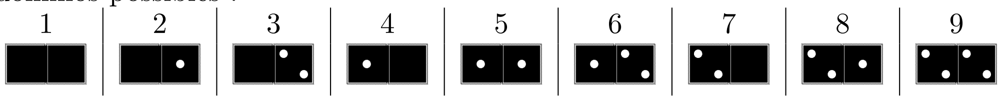
(*Suggestion : on peut traiter le cas $a \leq b \leq c$ et généraliser rapidement aux autres cas.*)

2 Domino

Le jeu des dominos est joué avec des pièces rectangulaires. Chaque pièce t à un nombre à gauche $l(t)$ et un à droite $r(t)$. Dans le jeu original chaque nombre (gauche ou droite) est un entier compris entre 0 et 6. La valeur 0 étant souvent appelé *joker*.

2.1 Dominos binaires avec un joker

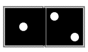
Pour simplifier nous considérons seulement comme nombre gauche ou droite les nombres 0 (joker), 1 et 2. Il y a alors 9 dominos possibles :



Un séquence $d_1 \cdots d_n$ de dominos est dite *valide* si pour chaque $1 \leq i < n$ soit le nombre de droite de d_i est égal au nombre gauche de d_{i+1}, ou que l'un des deux est un joker, formellement : $r(d_i) = l(d_{i+1})$ ou $0 \in \{r(d_i), l(d_{i+1})\}$).

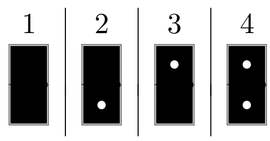
2.1. Montrer que le langage des séquences valides de dominos est régulier.

La valeur d'un domino est soit la somme de ses nombres droit et gauche s'ils ne sont pas nuls ou toutes les valeurs qui peuvent être obtenues en remplaçant chaque joker par 1 ou 2.

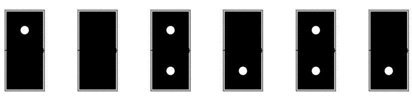
Par exemple,  a pour valeur 3 tandis que  vaut 3 et 4.

2.2. Montrer que le langage des séquences de dominos dont la somme peut être paire est régulier.

2.2 Dominos binaires



On s'intéresse maintenant aux dominos verticals dont les nombres sont 0 et 1 seulement. (La valeur 0 n'est plus un joker et est traitée comme un entier normal.) Il y a alors 4 dominos (voir à droite).



Quand on lit une séquence de dominos de gauche à droite on obtient deux nombres binaires –un pour la ligne du haut et un pour celle du bas– où les bits de poids forts sont lus en premier. Par exemple dans la séquence à gauche on lit $42 = 2^5 + 2^3 + 2^1$ dans la ligne du haut et $15 = 2^3 + 2^2 + 2^1 + 2^0$ dans celle du bas.

★ 2.3 Montrer que l'ensemble des séquences dans lesquelles le nombre du haut vaut trois fois celui du bas forme un langage régulier.

3 La propriété de dichotomie

Soit L un langage régulier accepté par un automate déterministe dont les états sont Q.

3.1. Montrer que L est non vide si et seulement si $\exists w \in L \ : \ |w| \leq |Q|$.

3.2. Montrer que L est infini si et seulement si $\exists w \in L \ : \ |Q| < |w| \leq 2|Q|$. (*Indice : Adapater la preuve du lemme de pompage avec un unique cycle.*)

4 Intersection d'un langage régulier et d'un langage algébrique

4.1. Soit L un langage régulier et L' un langage algébrique tous les deux sur l'alphabet Σ. Le langage $L \cap L'$ est-il algébrique? (*Indice : considérer les deux langages comme acceptés par des automates.*)

5 Expressions booléennes

On définit le langage des *expressions booléennes bien formées* (EBBF) par la grammaire $G := (\{Be, St\}, \Sigma, R, Be)$, où $\Sigma = \{\wedge, \neg, \top, \bot, (,)\}$ et les règles de production R sont $Be \to St\wedge St \mid \neg St \mid St$ et $St \to \top \mid \bot \mid (Be)$. Cette grammaire peut être prouvée non ambiguë et il existe donc un unique arbre de dérivation pour chaque terme. Ainsi à une EBBF on peut associer une valeur qui est vrai ($\top$) ou faux ($\bot$). (Le symbole $\wedge$ représente le ET logique tandis que $\neg$ représente la négation.)

5.1. Donner une grammaire produisant les termes correspondants à des EBBF valant *vrai*.

5.2. Donner un automate à pile acceptant les termes correspondants EBBF valant *faux*.

★ 5.3 Montrer que la grammaire des EBBF est non-ambiguë. (*Indice : Montrer que les mots produits par Be et St sont bien parenthèsés et regarder la première dérivation du plus petit mot qui a deux dérivations.*)

6 Les langages algébriques finis

6.1. Montrer que si tous les sous-ensemble d'un langage L sont algébriques alors L est fini. (*Indice : Construire récursivement un ensemble dont les mots grandissent vite et appliquer le lemme de pompage.*)

7 Automate universel

Un automate déterministe D peut être encodé de manière non ambiguë en une chaîne de caractères. (Par exemple, un automate $D := (Q, \Sigma, \delta, q_0, F)$ avec $\Sigma := \{0, 1\}$, $F \subseteq Q \subset \{0, 1\}^*$ et $q_0 = 0$ peut être encodé sur l'alphabet $\{0, 1, \#, (,)\}$ comme une liste de chaîne $(q\#b\#q')$ où $\delta(q, b) = q'$; les détails exacts de l'encodage n'étant pas importants, juste l'existence d'un tel encodage).

7.1. Existe-t-il un automate déterministe universel ? Plus précisément, si on choisit une fonction d'encodage f et que l'on regarde le langage: $L_U := \{f(D)\#w : w \in L(D)\}$, ce langage est-il régulier ?

8 Langages Unaires

8.1. Trouver tous les langages unaires réguliers. Pour cela, montrer qu'un langage régulier $L \subseteq 0^*$ peut être écrit comme l'union d'un langage fini et de langages $L_{b_i,c} := \{0^{b_i+cn} : n \geq 0\}$ pour $b_1, \ldots, b_k, c \in \mathbb{N}$. Inversement montrer que tous ces langages sont réguliers.

★★ 8.2 Trouver tous les langages algébriques unaires. Montrer qu'un langage unaire est algébrique si et seulement s'il est régulier. (*Suggestion : appliquer le lemme de pompage pour trouver la forme des mots longs. Et écrire le langage comme union de langages réguliers.*)