UNIVERSITÉ DE MONTRÉAL

DEVOIR 3

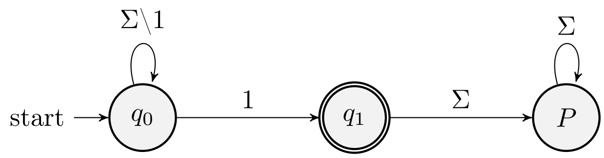
PAR
CHENGZONG JIANG (20122046)
MICHAEL PLANTE (20182677)
VANESSA THIBAULT-SOUCY (20126808)
JAYDAN ALADRO (20152077)
SOUKAINA BENABID (20148642)

BACCALAURÉAT EN INFORMATIQUE
FACULTÉ DE L'ÉDUCATION PERMANENTE

TRAVAIL PRÉSENTÉ À GENA HAHN
DANS LE CADRE DU COURS IFT 2105
INTRODUCTION À L'INFORMATIQUE THÉORIQUE

8 FÉVRIER 2021

Question 1



Soit $M = (Q, \Sigma, \delta, s, F)$ un AFD qui reconnaît $L_{m,1}$. Premièrement, il faut prouver que 1 $mod(m)$ est toujours 1. La formule pour le modulo est : $k \ mod \ m = k - \lfloor k/m \rfloor * m$. Vue que k est entre $0 \leq k < m$ le plancher de k/n est automatiquement 0, puisque $0 * m = 0$, il reste $k - 0$. Donc, lorsque $k = 1$, il va toujours rester 1. Le langage accepte donc seulement les valeurs binaires qui sont équivalentes à 1. Pour prouver que ce langage est régulier, il faut pouvoir construire un AFD qui reconnaît se langage. Preuve par récurrence :

1. $Q = \{q_0, q_1, P\}$
2. $\Sigma = \{0, 1\}$
3. $\delta(q_0, 0) = q_0$
   $\delta(q_0, 1) = q_1$
   $\delta(q_1, \Sigma) = P$
   Il faut montrer que : $\hat{\delta}(q_0, w) \in F \longleftrightarrow w \in L$
4. $s = q_0$ est l'état initial
5. $F = \{q_1\}$

Le cas de base est le mot vide :

$$\hat{\delta}(q_0, w) \mid w = \varepsilon, \ \ \hat{\delta}(q_0, w) = q_0 \tag{1}$$

Comme $q_0 \notin F$, il est vrai que l'AFD que nous avons construit rejette le mot vide. Le mot vide ne faisant pas partie de notre langage, le cas de base est vrai.

Pour $\delta(q_0, w)$ où $w = vx$, $|v| \geq 1$ et $x = 1$. Preuve par récurrence pour la longueur de v :

— Si $v = 0$ ($|v| = 1$) :

$$\begin{aligned} \hat{\delta}(q_0, w) &= \delta((\delta(q_0, 0), 1) && (2) \\ &= \delta(q_0, 1) && (3) \\ &= q_1 && (4) \end{aligned}$$

Lorsque $v = 0$ et $x = 1$, le mot va être accepté. Le cas de base est vérifié.

— Si $v = 1$ ($|v| = 1$) :

$$\begin{aligned} \hat{\delta}(q_0, w) &= \delta((\delta(q_0, 1), 1) && (5) \\ &= \delta(q_1, 1) && (6) \\ &= P && (7) \end{aligned}$$

Le cas de base lorsque $v = 1$ est rejeter par l'AFD. Donc, peut importe les valeurs du mot, s'il y a plus qu'un 1, le mot ne va pas être accepté. L'AFD va directement dans l'état poubelle.

Pour le cas générale, on suppose que $|v| = k$ est accepté. Il faut prouver que $|v| = k+1$ est vrai aussi. Pour cette preuve, le cas de base a prouvé qu'un mot ne peut pas avoir plus qu'un 1 pour être accepté. Comme $x = 1$ pour cette preuve, nous allons nous concentré sur le cas où v à un $k+1$ nombre de 0. k nombre de 0 avec $x = 1$ est vrai par hypothèse. Ce qui implique que $\hat{\delta}(q_0, w)$ est accepté. Pour que le mot soit accepté, il faut que $\delta(\hat{\delta}(q_0, v), x) = q_1$. Vue que $q_1 \in F$. De plus, il n'y a qu'une transition vers q_1. C'est lorsque l'AFD lis un 1. Par ce fait même, $\hat{\delta}(q_0, v)$ (lorsque $|v| = k$) doit être égal à q_0. Il faut donc prouver que $k+1$ nombre de 0 va toujours être dans l'état initial :
Lorsque $|v| = k+1$, où $v = s0$ et $s = 0_1 0_2 ... 0_k$

$$\begin{aligned}
\hat{\delta}(q_0, v) &= \delta((\hat{\delta}(q_0, s), 0) && (8)\\
&= \delta(q_0, 0) && (9)\\
&= q_0 && (10)
\end{aligned}$$

Par hypothèse, $\hat{\delta}(q_0, s) = q_0$. De plus, $\delta(q_0, 0) = q_0$. Donc, la preuve par récurrence prouve que n'importe quelle nombre de 0 avant le 1 est accepté par l'AFD. Par le cas de base, aussitôt qu'il y a plus qu'un 1, l'AFD reste dans l'état poubelle.

Maintenant, pour $x = 0$. Il faut aussi faire une preuve par récurrence. Le cas de base est lorsque $|v| = 1$. $v = 0$:

$$\begin{aligned}
\hat{\delta}(q_0, w) &= \delta(\delta(q_0, 0), 0) && (11)\\
&= \delta(q_0, 0) && (12)\\
&= q_0 && (13)
\end{aligned}$$

Le cas de base n'est pas acceptant, car l'état final est $q_0 \notin F$

$v = 1$:

$$\begin{aligned}
\hat{\delta}(q_0, w) &= \delta(\delta(q_0, 1)0) && (14)\\
&= \delta(q_1, 0) && (15)\\
&= P && (16)
\end{aligned}$$

Le cas de base n'est pas acceptant, car l'état final est $P \notin F$. Les deux cas de base ne sont pas acceptant. Ceci implique que si le mot fini avec un 0, il ne sera pas accepté.

Lorsque l'AFD lit un mot w, s'il lit une entrée alors qu'il se trouve dans l'état acceptant q_1, il se dirigera dans l'état P et ce mot ne sera jamais accepté par l'AFD. C'est pourquoi si le nombre d'entrées "1" est plus grand que 1, le mot ne sera pas accepté. La seule façon que notre AFD accepte un mot est lorsque ce mot possède un nombre k de "0" et que son dernier symbole soit exclusivement un seul "1". Dans le cas contraire, il ne sera pas accepté. Le langage est donc régulier, puisqu'il existe un AFD qui le reconnaît. L'AFD fonctionne et accepte seulement les mots binaires ayant une valeur décimale de 1.

Par preuve par récurrence, $L_{m,1} = \{w \in \{\Sigma^* : n(w) \equiv 1 \ (\mathrm{mod} m)\}$ est régulier.

Question 3

Supposons que le langage $L = \{w \in \{a, b, c, d\} \mid |w|_{ab} = |w|_{ba}\}$ est régulier. Soit $p \in \mathbb{N}^{\geq 0}$ la longueur de pompage. Choisissons $w = a^p b^p a^p c^p d^p$. On a $|w| = 5p > p$ et $w \in L$ car : $\forall p \geq 1$ soit $|w|_{ab} = 1$ et $|w|_{ba} = 1$.

Le lemme de pompage garantit qu'il y a un w qui peut être séparé en trois morceaux, $w = xyz$ où $\forall i \geq 0$ $xy^i z \in L$, $|y| > 0$ et $|xy| \leq p$. Décomposons :

Comme $|xy| \leq p$ alors $x = a^n$ et $y = b^m a^m c^m$ avec $n + 3m \leq p$, $m \geq 0$ et $n \geq 0$
Puis $z = d^p$

Nous avons alors $xy^i z = a^n (b^m a^m c^m)^i d^p$.
Prenons $i = 2$:

$xy^2 z = a^n (b^m a^m c^m)^2 d^p \longrightarrow$ On a donc $|w|_{ab} = 1$ et $|w|_{ba} = i = 2$.

$\longrightarrow |w|_{ab} \neq |w|_{ba}$ Par contradiction, le lemme du pompiste ne fonctionne pas dans notre situation, donc L est un langage non-régulier.

Question 4

(a). Soit Σ un alphabet.On appelle langage sur Σ tout sous ensemble de Σ^* : $L \subseteq \Sigma^*$.
Epsilon (ϵ) représente le mot vide (de longueur 0). C'est donc un élément du langage L.
Ainsi, comme L est un ensemble et ϵ est un élément, il est faux de dire qu'un ensemble est égale à un élément.
Donc, l'énoncer $L = \epsilon$ n'a pas de sens.

(b). Par définition, un alphabet Σ est un ensemble fini non vide de symboles ou de lettres.
Or, $\{\}$ est un ensemble qui ne contient aucun élément (vide).
Donc l'énoncer : $\Sigma = \{\}$ n'a pas de sens.

(c). Posons Σ^+ comme l'ensemble de tous les mots de longueur supérieure ou égale à 1.
Par définition, Σ^* représente l'ensemble de tous les mots de toutes les longueurs, 0 y compris :
$\Sigma^* = \{\epsilon\} \cup \Sigma^+$. Donc, l'énoncer $\Sigma = \{\epsilon\}$ n'a pas de sens.

(d). Comme mentionné plus haut, un alphabet ne peut pas être vide, Donc $\emptyset$ ne peut pas appartenir à Σ. De plus, le mot vide ϵ ne peut pas non plus appartenir à Σ.
Donc l'énoncer : $\Sigma = \{\epsilon, \emptyset\}$ n'a pas de sens.

(e). Par définition :
$A \in B$ signifie que l'élément A est contenu dans l'ensemble B, soit A élément de l'ensemble B.
$A \subseteq B$ signifie que A est un ensemble qui est un sous ensemble de l'ensemble B, soit tous les éléments de l'ensemble A sont dans l'ensemble B.
Supposons que $L \subseteq \Sigma^*$ revient à $L \in \Sigma^*$.
Définissons l'alphabet $\Sigma = \{\epsilon, a, b\}$ et le langage $L = \{\epsilon, a, bb\}$.
Tous les éléments de l'ensemble L sont dans l'ensemble Σ. Mais, $\{\epsilon, a, bb\}$ n'est pas un élément de $\{\epsilon, a, b\}$.
L'ensemble $\Sigma = \{\epsilon, a, b\}$ contient les éléments ϵ, a et b et non pas l'élément : $\{\epsilon, a, bb\}$. Donc, $L \subseteq \Sigma^*$ n'est pas équivalent à $L \in \Sigma^*$ Donc l'énoncer : $L \in \Sigma^*$ n'a pas de sens.

(f). L'ensemble des entiers naturels $\mathbb{N}$ représente l'ensemble des entiers naturels. Cet ensemble est infini, or par définition, un alphabet Σ est fini donc l'énoncer n'a pas de sens.

(g). Un mot est une suite, séquence, finie et ordonnée d'éléments de l'alphabet.
Or, $\{abcaa\}$ est un ensemble et non une suite. Donc cet énoncé n'a pas de sens.

(h). Le langage vide : $L^0 = \{\epsilon\}$ contient un élément : le mot de longueur 0. Donc cet ensemble n'est pas vide. L'énoncer est absurde.

(i). Par définition, L^k est le langage de longueur k, soit l'ensemble de mots de longueur k symboles de l'alphabet Σ.
L^0 est l'ensemble de mot de longueur 0, soit l'ensemble contenant le mot vide ϵ.
Donc, l'énoncer $L^0 = \{\epsilon\}$ a du sens.

(j). L'ensemble contenant le mot vide est différent de l'élément mot vide : $\{\epsilon\}$ n'est pas égale à ϵ. Donc l'énoncer n'a pas de sens.

(k). $\epsilon \in L^*$ est vrai. Et nous avons montré avant que l'ensemble $\{\epsilon\}$ n'est pas égale à ϵ c'est plutôt $\{\epsilon\} \subseteq L^*$ car $\{\epsilon\} \cup L^* = L^*$.
Donc, l'énoncer $\epsilon \subseteq L^*$ n'a pas de sens car on ne peut pas faire l'union d'un élément et d'un ensemble (Absurde).