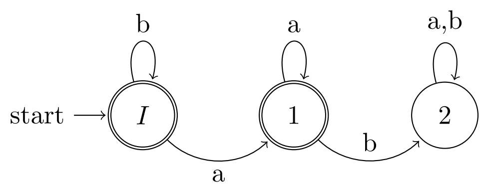
TP1 - IFT2105

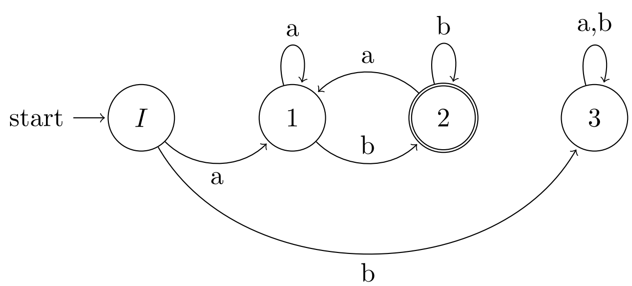
par Ilan Elbaz

13 Mai 2019

1. Pour les deux automates M1 et M2, répondez aux questions suivantes.



M1



M2

1. Quelle est la définition formelle de l'automate?
2. Quels sont les états acceptants ?
3. Quelle séquence d'états la machine traverse-t-elle pour l'entrée *aabab* ?
4. La machine accepte-t-elle la chaîne *aabab* ?
5. La machine accepte-t-elle la chaîne vide ϵ ?
6. Quel language est reconnu par la machine?

RÉPONSES:
Pour M_1:

1. Quelle est la definition formelle de l'automate?
   Soit $M_1 = (Q, \Sigma, \delta, s, F)$ défini par :
   - $Q = \{I, 1, 2\}$ l'ensemble fini d'états;
   - $\Sigma = \{a, b\}$ l'alphabet;
   - δ est donné par la table de transition suivante:

	a	b
I	1	I
1	1	2
2	2	2

   - $Q_0 = I$ l'état initial ;
   - $F = \{I, 1\}$ l'ensemble des états acceptants;

2. Quels sont les états acceptants ?
   Les états acceptants sont $F = \{I, 1\}$.

3. Quelle séquence d'états la machine traverse-t-elle sur entrée $aabab$?

$$\begin{aligned} [aabab, I] &\vdash [abab, 1] \\ &\vdash [bab, 1] \\ &\vdash [ab, 2] \\ &\vdash [b, 2] \\ &\vdash [\epsilon, 2] \end{aligned}$$

4. La machine accepte-t-elle la chaîne $aabab$?
   Non car l'état 2 n'est pas acceptant.

5. La machine accepte-t-elle la chaîne vide ϵ ?
   Oui car l'état initial I est acceptant.

6. Quel language est reconnu par la machine?
   $L1 \subseteq \{a, b\}^*$ défini par $L_1 = \{w : w \text{ ne contient pas la séquence } ab\}$

Pour M_2:

1. Quel est la definition formel de l'automate?
   Soit $M_2 = (Q, \Sigma, \delta, s, F)$ défini par :

   - $Q = \{I, 1, 2, 3\}$;
   - $\Sigma = \{a, b\}$;
   - δ est donné par la table de transition suivante:

	a	b
I	1	3
1	1	2
2	1	2
3	3	3

   - $q_0 = I$;
   - $F = \{2\}$;

2. Quels sont les états acceptants ?
   L'état acceptant est $F = \{2\}$.

3. Quelle séquence d'états la machine traverse-t-elle sur entrée $aabab$?

$$\begin{aligned}[aabab, I] &\vdash [abab, 1]\\ &\vdash [bab, 1]\\ &\vdash [ab, 2]\\ &\vdash [b, 1]\\ &\vdash [\epsilon, 2]\end{aligned}$$

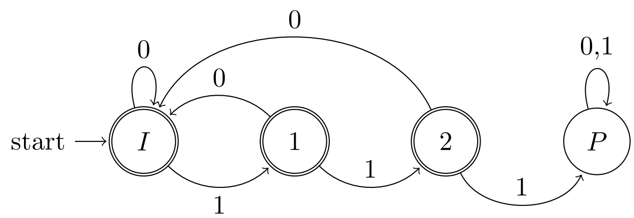
4. La machine accepte-t-elle la chaîne $aabab$?
   oui car l'état 2 est acceptant.

5. La machine accepte-t-elle la chaîne vide ϵ ?
   Non I est non acceptant.

6. Quel language est reconnu par la machine?
   $L1 \subseteq \{a, b\}^*$ défini par
   $L_1 = \{w : ax_1 \cdots x_n b$ avec $n \geq 0$ et $x_i \in \Sigma = \{a, b\}\}$

2. Montrez que le langage L des mots binaires ne contenant pas plus de deux 1 consécutifs est régulier.

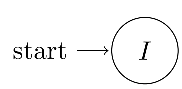
Pour montrer qu'un langage est régulier, on montre qu'il est reconnu par un automate à état fini déterministe avec une explication du rôle de chaque état. On donne un automate qui reconnaît L.



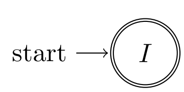
- I : L'état initial dans lequel va débuter l'algorithme
- 1 : état qui indique qu'un "1" vient d'être lu
- 2 : état qui indique que deux "1" à la suite viennent d'être lus
- P : état "poubelle" qui indique que la chaîne est composer de "11". Peu importe les prochains caractères lus on restera dans cet état non acceptant.

3. Montrez à l'aide d'automates que les langages $L_1 = \emptyset$ (langage vide) et $L_2 = \{\epsilon\}$ sont réguliers.

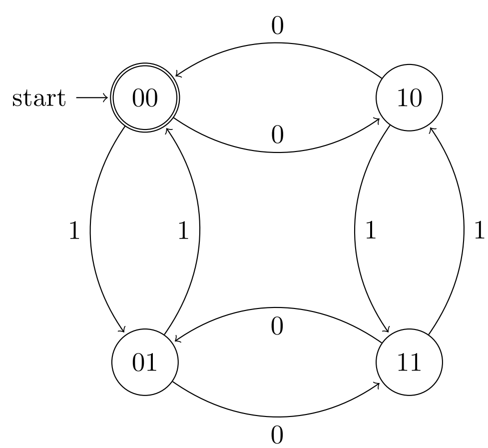
Pour $L_1 = \{\emptyset\}$, on donne un automate qui ne reconnaît aucun mot.



Pour $L_2 = \{\epsilon\}$, on donne un automate qui reconnaît juste le mot vide.



4. Soit L le language qui contient des mots avec un nombre pair de 0 et de 1. Donnez un automate qui décide de L. Décrire de manière formelle le langage.



$L = \{w \in \{0,1\}^* \mid |w|_a = 2n,\ |w|_b = 2m,\ n \in \mathbb{N},\ m \in \mathbb{N}\}$

5. Pour chacun des langages décris, donner l'écriture formelle du langage.

On assume que les langages suivants sont définis sur l'alphabet $\Sigma = \{a, b\}$

- L_1 contient tous les mots qui sont constitué d'au plus cinq a.
  $L_1 = \{w \mid |w|_a < 6\}$
- L_2 contient tous les mots qui ont deux fois plus de a que de b.
  $L_2 = \{w \mid |w|_a = 2 * |w|_b\}$
- L_3 contient tous les mots qui de taille impaire avec un b au milieu.
  $L_3 = \{w \mid w = x \cdot b \cdot z,\ |x| = |z|\}$
- L_4 contient tous les mots de taille 1 et plus avec un nombre impaire de b.
  $L_4 = \{w \mid |w| > 0,\ |w|_b = 2k + 1, k \in \mathbb{N}\}$
- L_5 contient tous les mots de taille 2 avec deux a au minimum.
  $L_5 = \{w \mid |w|_a > 2\}$

6. Décrire brièvement chacun des langages ci-dessous.

On assume que les langages suivants sont définis sur l'alphabet $\Sigma = \{0, 1\}$

- $L_1 = \{w \mid w \mod 2 = 0\}$
  L_1 est le langage comprenant tous les mots binaires pairs.
- $L_2 = \{w \mid |w|_0 = 1, |w|_1 = 2\}$
  L_2 est le langage comprenant tous les mots binaires composé de un 0 et deux 1.
- $L_3 = \{w \mid |w|_0 + |w|_1 = 3k, k \in \mathbb{N}\}$
  L_3 est le langage comprenant tous les mots binaires dont la taille est un multiple de 3.
- $L_4 = \{w \mid |w = 1100 \cdot x_1 \cdots x_n \cdot 0011,\ n \in \mathbb{N}\}$
  L_4 est le langage comprenant tous les mots binaires commençant par "1100" et finissant par "0011" (sans chevauchement).
- $L_5 = \{w \mid w = w^R\}$
  L_5 est le langage des mots binaires palindromes.

7. Donner l'automate fini déterministe reconnaissant les entiers binaires qui sont des multiples de 3.

Ici on va construire un automate reconnaissant les entiers binaires qui sont des multiples de 3.
L'automate aura exactement 3 états (s0,s1,s2) représentant les trois résultat possible de la fonction modulo3(X):

- X mod 3 = 0 (s0 état acceptant des multiples de 3)
- X mod 3 = 1 (s1)
- X mod 3 = 2 (s2)

Pour déterminer δ et construire diagramme de transition de l'automate, il suffit d'étudier l'effet qu'a la lecture d'un 1 ou d'un 0 sur le modulo du nombre lu à l'état courant.

Ainsi, de l'état s0, on a:

- 1_2 mod $3 = 1$, transition vers s1 sur lecture d'un 1.
- 0_2 mod $3 = 0$, transition vers s0 sur lecture d'un 0.

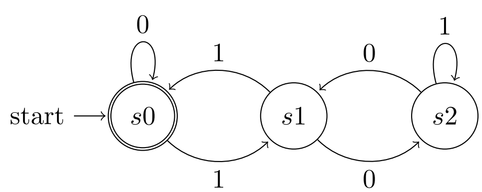
En poursuivant la lecture de l'état s1, on a:

- 11_2 mod $3 = 1$, transition vers s0 sur lecture d'un 1
- 10_2 mod $3 = 0$, transition vers s2 sur lecture d'un 0.

En poursuivant la lecture de l'état s2, on a:

- 101_2 mod $3 = 2$, transition vers s2 sur lecture d'un 1
- 100_2 mod $3 = 1$, transition vers s1 sur lecture d'un 0.

On obtient alors l'automate ci-dessous:



8. Démontrer que la classe REG est fermée pour l'intersection.

Preuve : Soient $K1 \in REG$ et $K2 \in REG$ deux langages réguliers reconnus respectivement par les automates M_1 et M_2:

$$\begin{aligned} M_1 &= (Q_1, \Sigma, \delta_1, q_{0,1}, F_1) \\ M_2 &= (Q_2, \Sigma, \delta_2, q_{0,2}, F_2) \end{aligned}$$

Considérons l'automate M suivant :

$$M = (Q_1 \times Q_2, \Sigma, \delta, (q_{0,1}, q_{0,2}), F_1 \times F_2)$$

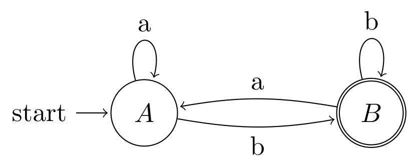
où :

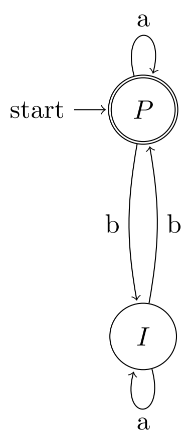
$$\delta((q_1, q_2), x) = (q'_1, q'_2)$$

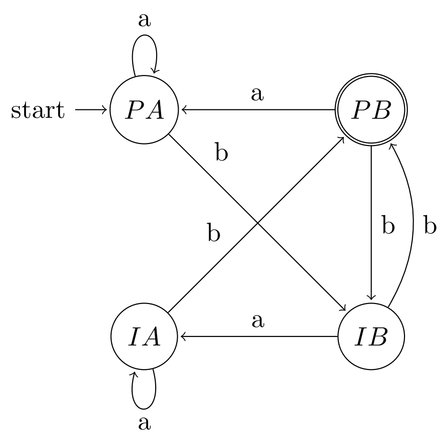
si et seulement si

$$\delta_1(q_1, x) = q'_1 \text{ et } \delta_2(q_2, x) = q'_2$$

comme dans l'exemple ci dessous:







Soit $w \in K1 \cap K2$.
Alors w amène à M_1 de $q_{0,1}$ à un état final $f_1 \in F_1$ et amène également M_2 de $q_{0,2}$ à un état final $f_2 \in F_2$.
Par construction, le mot w amène l'automate M de $(q_{0,1}, q_{0,2})$ à $(f_1, f_2) \in F_1 \times F_2$. En conséquence, M accepte w et $w \in L(M)$.
A contrario, si $f1 \notin F_1$ ou $f_2 \notin F_2$, alors $(f_1, f_2) \notin F_1 \times F_2$ et M n'accepte pas w.
On a donc $L(M) = K_1 \cap K_2$ avec $K_1 \cap K_2 \in REG$.

Lemme du pompiste - Rappel

Soit $L \in REG$. Le lemme du pompiste dit la chose suivante :
Il existe une longueur de pompage $p > 1$ telle que pour tout mot $w \in L$ de longueur au moins p, w peut être écrit par $w = xyz$ avec

1. $\mid y \mid \geq 1$
2. $\mid xy \mid \leq p$
3. $\forall i \geq 0,\ xy^iz \in L$

9. Prouvez à l'aide du théorème du pompiste que L sur $\Sigma = \{a, b\}$ défini par $L = \{x \cdot a^n \mid x \in \Sigma^* \text{ et } |x| = n\}$ n'est pas régulier.

Supposons au contraire que $L \in REG$, afin d'arriver à une contradiction.

Soit $p \geq 1$ tel que donné par le lemme du pompiste.
Considérons le mot $w = b^p a^p$ On a bien que $w \in L$ et $\mid w \mid = 2p \geq p$.

Soit $xyz = w$ une décomposition quelconque en trois parties de w. Supposons que les deux premières conditions du lemme tiennes, soient $|xy| \leq p$ et $\mid y \mid \geq 1$ montrons que la troisième condition ne peut pas tenir.

Si $|xy| \leq p$ et $|y| \geq 1$, alors y contient uniquement des b. Observons le mot donné lorsqu'on pompe une fois : $xy^2z = b^{p+|y|}a^p$. Étant donné que ce mot contient strictement plus de b que de a, celui-ci ne peut pas être dans L. On arrive à une contradiction et on conclut que $L \notin REG$.

10. Prouvez que L sur $\Sigma = \{a, b\}$ défini par $L = \{xx \mid x \in \Sigma^*\}$ n'est pas régulier.

Si L est régulier, alors le lemme du pompiste doit tenir
Soit $p \geq 1$.
Prenons le mot $ww = a^p b^p a^p b^p \in L$ séparable en trois parties xyz

Avec $|xy| \leq p$ et $|y| \geq 1$ on a que y ne contient que des a. Or $xy^3z \notin L$ car il est de la forme $a^{p+2|y|}b^p a^p b^p$. En effet on observe que la première moitié du mot contient plus de a que la deuxième. L n'est donc pas régulier

11. Montrez que L' sur $\Sigma = \{a, b\}$ défini par $L' = \{xy \mid |x| = |y| \text{ et } x \neq y\}$ n'est pas régulier.

L est constitué des mots de longueur paire qui ne sont pas constitués de deux mots identiques.
Regardons plutôt les langages $L = \{xy \mid |x| = |y| \text{ et } x = y\}$ et $M = \{x \mid |x| = 2n$ avec $n \in \mathbb{N}\}$. On a donc:

$$L = M \cap \overline{L'}$$

Or on à déjà montré que L n'etait pas régulier dans les questions précédentes. Si M est régulier, alors on conclu que L' ne l'est pas puisque la classe REG est fermée sous la complémentation et sous l'intersection.
En effet, si $M \in REG$ et $L' \in REG$, alors $\overline{L'} \in REG$ et $L = M \cap \overline{L'} \in REG$ ce qui amène a une contradiction.