Théorie des Langages

Série 2

LES AUTOMATES A ETATS FINIS & LANGAGES REGULIERS

Exercice 6 :

1) $L_1 = \{a^n b^p \ / \ n\text{-}p = 3k \ / \ k \in Z\}$

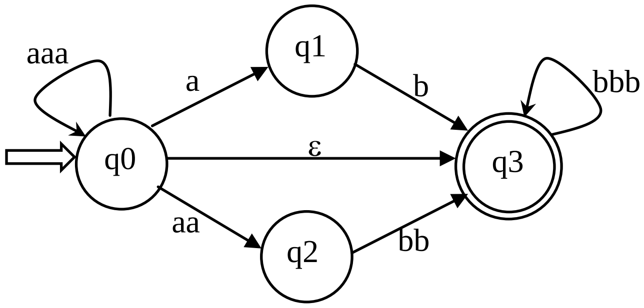
Exemple : $L_1=\{\varepsilon$, ab, aabb, aaa, bbb, aaaab, abbbb, aaabbb, abbbbbbb, … }

Remarque : Les mots de L_1 sont composés d'une suite de **a** suivie d'une suite de **b** où la différence entre le nombre de **a** et le nombre de **b** est un multiple de 3 (Le nombre de **a** peut être supérieur ou inférieur au nombre de **b** car $k \in Z$, i.e. **k** peut positif, négatif ou nul).

On a : $n\text{-}p=3k \ / \ k \in Z$ ssi ($n=3i$ et $p=3j$) ou ($n=3i+1$ et $p=3j+1$) ou ($n=3i+2$ et $p=3j+2$). (Il faudrait que le nombre de **a** et le nombre de **b** aient le même reste de la division sur 3).

Donc, dans les mots de ce langage, il faut que le nombre de **a** et le nombre de **b** soient dans l'une des combinaisons suivantes : (3i , 3j) ou (3i+1 , 3j+1) ou (3i+2 , 3j+2). La première composante de chaque couple représente le nombre de **a** et la deuxième composante représente le nombre de **b**.

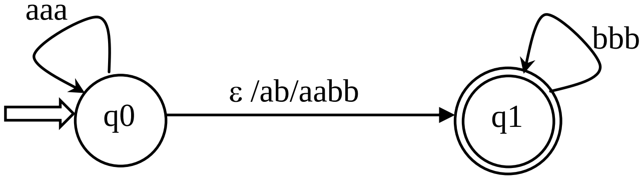
D'où l'automate :



C'est un automate généralisé.

Explication de l'automate : Dans l'état q1, le nombre de **a** est 3i+1 et en transitant vers q3 par **b** le nombre de **b** dans l'état q3 serait 3j+1. Par contre, à l'état q2, le nombre de **a** est 3i+2 et en transitant vers q3 par **bb** le nombre de **b** dans l'état q3 serait 3j+2. Dans l'état q0, le nombre de a est 3i et en transition par le mot vide vers l'état q3, le nombre de b dans l'état q3 serait 3j.

L'automate précédent peut être compacté comme suit :



Remarque : Le langage L_1 peut s'écrire comme suit : $\{a^{3i}wb^{3j} \ / \ w=\varepsilon$ ou ab ou aabb et $i, j \geq 0\}$.

Théorie des Langages

Série 2

LES AUTOMATES A ETATS FINIS & LANGAGES REGULIERS

2) Soit $MIL(L_1) = \{w \mid \exists u, v \in \{a, b\}^* \text{ et } uwv \in L_1\}$

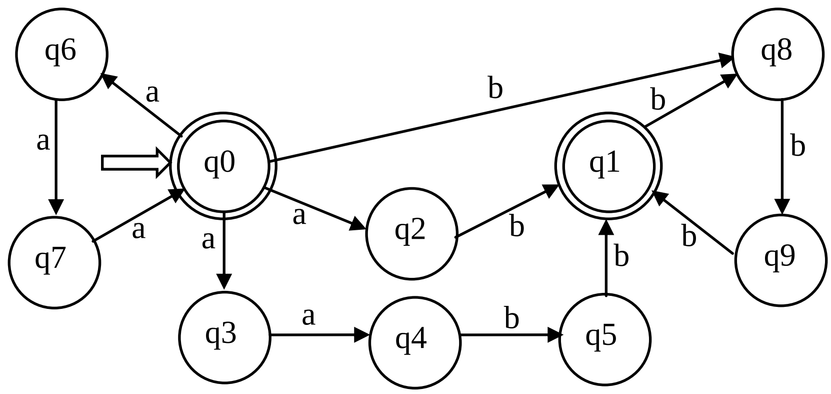
Remarque : $MIL(L_1)$ représente tous les milieux des mots de L_1. En d'autres termes, les sous mots ou les facteurs des mots de L_1.

La forme générale des mots de L_1 est la suivante : $a_1 \ldots a_k \ldots a_n b_1 \ldots b_l \ldots b_p$. où pour chaque a_i / $i \in [1,n]$ et chaque b_j / $j \in [1,p]$ et $n-p = 3k$ et $k \in Z$. Ainsi, les **sous mots** des mots de L_1 sont composés d'une suite aléatoire (éventuellement vide) de **a** suivie d'une suite aléatoire (éventuellement vide) de **b**.

Donc, $MIL(L_1) = \{ a^i b^j \mid i, j \geq 0\}$

3) Démontrons que le langage $MIL(L_1)$ est régulier.
Pour démontrer que $MIL(L_1)$ est régulier, il suffit de trouver un automate d'état fini le reconnaissant. Le but de cette question est de construire cet automate à partir de celui de L_1 afin de généraliser le résultat pour un langage régulier quelconque. Notons que l'automate reconnaissant L_1 proposé dans la question précédente est généralisé et ne permet pas d'avoir l'automate de MIL(L1).

Construisons tout **d'abord l'automate simple** reconnaissant le langage L_1.



La construction de l'automate reconnaissant $MIL(L_1)$ se fait à partir de l'automate simple reconnaissant L_1 comme suit :

Les mots de $MIL(L_1)$ sont des sous mots des mots de L_1 qui sont eux même de la forme $\mathbf{a_1 \ldots a_k \ldots a_n b_1 \ldots b_l \ldots b_p}$ Donc on commencera la lecture des mots de $MIL(L_1)$ à partir de n'importe quel symbole des mots de L_1. Ainsi, on crée un nouvel état initial qui sera relié à tous les états de l'automate de L_1 (car on ne peut pas avoir plusieurs états initiaux).

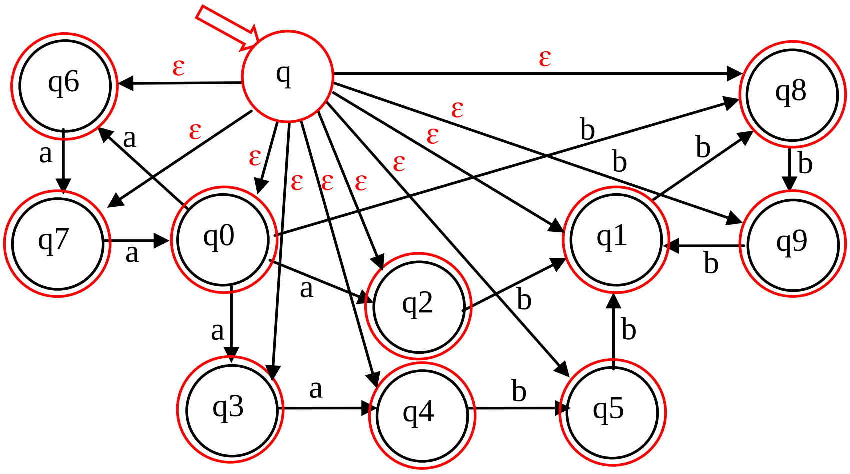
Les mots de $MIL(L_1)$ se terminent avec n'importe quel symbole dans les mots de L_1, donc tous les états de L_1 deviennent des états finaux.

Théorie des Langages

Série 2

LES AUTOMATES A ETATS FINIS & LANGAGES REGULIERS

Ainsi, l'automate reconnaissant MIL(L_1) est construit à partir de celui reconnaissant (L_1) comme suit :



D'une manière générale, soit l'automate simple et émondé $A_1=(X_1, Q_1, q0_1, \delta_1, F_1)$ reconnaissant un langage L_1. L'automate $A=(X, Q, q0, \delta, F)$ reconnaissant le langage MIL(L_1) est défini comme suit :

$X=X_1$ /* L'alphabet de A est celui de A1*/
$Q=Q_1\cup\{q0\}$ $q0\notin Q_1$ /* Les états de A sont ceux de A1 plus un nouvel état initial */
q0 /* Le nouvel état initial*/
$F=Q_1$ /* Tous les états de A1 deviennent finaux*/
$\delta=\delta_1\cup\{\delta(q0,\varepsilon)=qi \ \ \forall qi\in Q_1\}$ /*On maintient toutes les transitions de A_1 et on leur ajoute des transitions reliant le nouvel état initial à tous les états de A_1 par epsilon*/

Remarque : Un automate d'états finis est émondé si tous ses états sont accessibles et co-accessibles