

Université de Technologie de Compiègne
AI21 - Théorie des Langages de Programmation
TD 5 Automates : Corrigé

Dans ce TD nous abordons les automates finis déterministes (AFD), les automates non déterministes (AFN) et les automates avec epsilon-transition (eps-AFD).

Exercice 1

Construire les automates d'états finis simples et déterministes acceptant les langages dénotés par les expressions régulières suivantes :

1. Soit l'expression régulière $E_1 = ab(a|b)^*$:

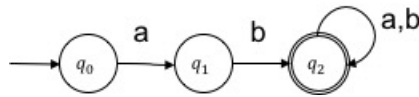


FIGURE 1 – Automate reconnaissant $L(E_1)$

2. Soit l'expression régulière $E_2 = 110^*1$:

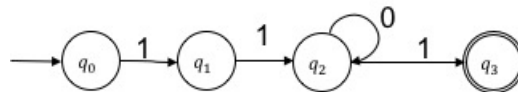


FIGURE 2 – Automate reconnaissant $L(E_2)$

3. Soit l'expression régulière $E_3 = aa^*|b$. L'automate reconnaissant le langage engendré par l'expression régulière E_3 est donné par la Figure 3. On rappelle

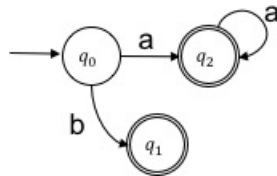


FIGURE 3 – Automate reconnaissant $L(E_3)$

que l'opérateur concaténation est plus prioritaire que l'opérateur de disjonction. Il ne faut confondre l'expression régulière E_3 avec l'expression régulière suivante : $E'_3 = a(a^*|b)$ où on force la priorité de l'opérateur de disjonction grâce aux méta symboles parenthèses.

On donne aussi, dans la Figure 4, l'automate reconnaissant le langage $L(E'_3)$ (non demandé dans l'exercice).

Construire des AFD, AFN ou eps-AFN acceptant les langages dénotés par les expressions régulières suivantes.

1. Soit l'expression régulière $E_4 = (a|b)^*ab(a|b)^*$. On remarque que l'automate de la Figure 5 est non déterministe.

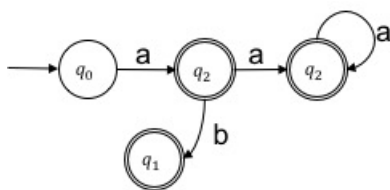


FIGURE 4 – Automate reconnaissant $L(E_3)$

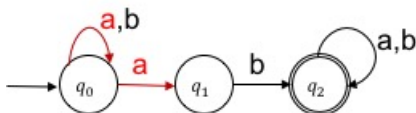


FIGURE 5 – Automate reconnaissant $L(E_4)$

2. Soit l'expression régulière $E_5 = (ab)^*(baa)^*aa$. On remarque que l'automate, de la Figure 6, contient des ϵ transition.

Remarque : Il est généralement plus naturel et plus facile de construire des automates non déterministes avec ϵ -transition. On rappelle qu'il existe une équivalence entre AFD, AFN et eps-AFN. A titre d'exemple on donne un AFD (Figure 7) qui reconnaît le langage $L(E_5)$.

D'une manière générale on peut utiliser l'algorithme vu en cours pour obtenir un automate fini déterministe.

Exercice 2

Construire les automates d'états finis simples et déterministes acceptant les langages suivants :

1. L'ensemble des constantes composées de cinq chiffres maximum.
Soit $C = [0 - 9]$ et $C_0 = [1 - 9]$.
 - La Figure 8 représente une solution possible où on autorise la présence d'un ou plusieurs zéros à gauche de la constante.
 - La Figure 9 représente une solution possible où on n'autorise pas la présence d'un ou plusieurs zéros à gauche de la constante.
2. L'ensemble des chaînes composées de lettres, de chiffres et de tirets. Elles ne peuvent ni commencer, ni finir par un tiret et ne contiennent pas deux tirets consécutifs.
Soit $c = [0 - 9]$ et $l = [a - zA - Z]$. L'automate est donné par la Figure 10, il accepte la chaîne vide. Dans le cas où la chaîne vide ne devrait pas être acceptée par l'automate, on peut rendre l'état q_0 non final.
3. L'ensemble de toutes les chaînes de 0 et de 1 telles que chaque 0 soit immédiatement suivi par au moins un 1.
Les automates des Figures 11 et 12 représentent deux solutions possibles. Le premier accepte la chaîne vide, alors que le deuxième l'interdit.
4. L'ensemble de chaînes formées de lettres telles que les consonnes sont séparées par des voyelles (pas de consonnes consécutives).
On représente par lc et lv les lettres consonnes et les voyelles, respectivement. Les automates des Figures 13 et 14 représentent deux solutions possibles. Le premier accepte la chaîne vide, alors que le deuxième l'interdit.

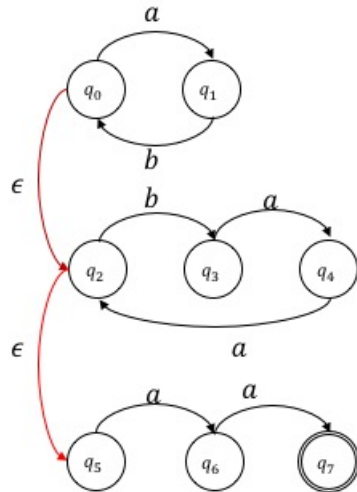


FIGURE 6 – Automate reconnaissant $L(E_5)$

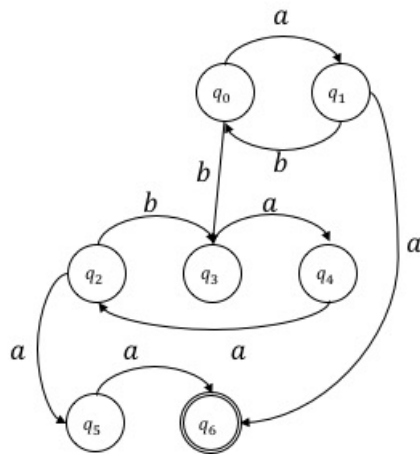


FIGURE 7 – Automate reconnaissant $L(E_5)$

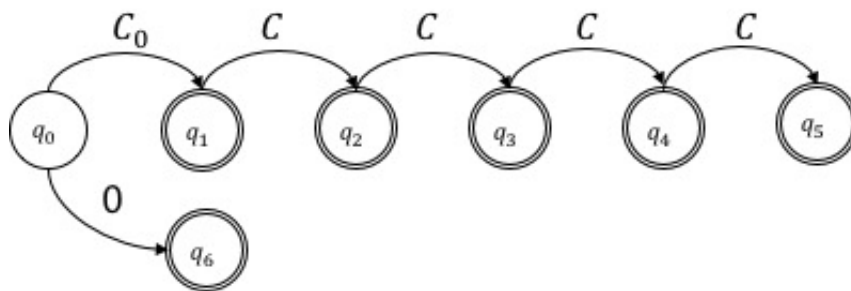


FIGURE 8 – Exercice 2-1 : Solution 1

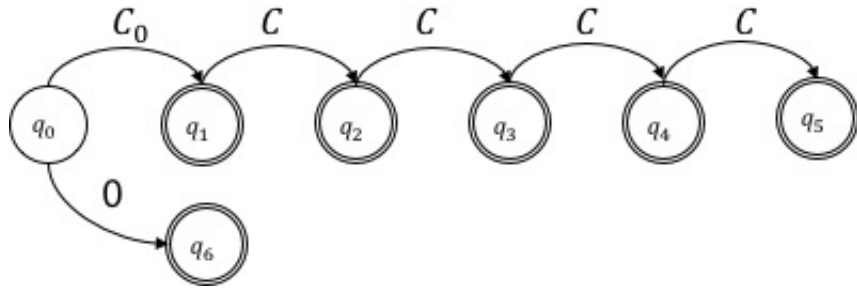


FIGURE 9 – Exercice 2-1 : Solution 2

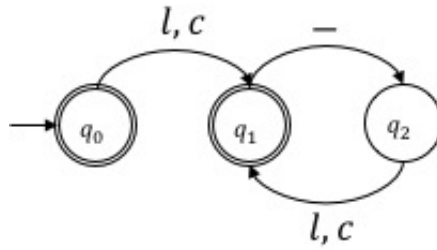


FIGURE 10 – Automate exercice 2-2

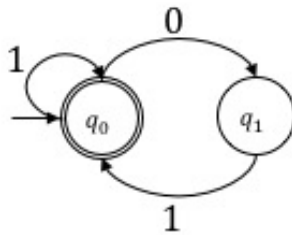


FIGURE 11 – Exercice 2-3 : Solution 1

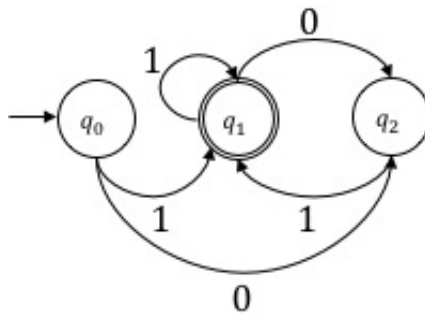


FIGURE 12 – Exercice 2-3 : Solution 2

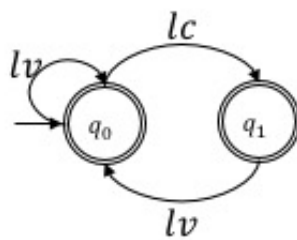


FIGURE 13 – Exercice 2-3 : Solution 1

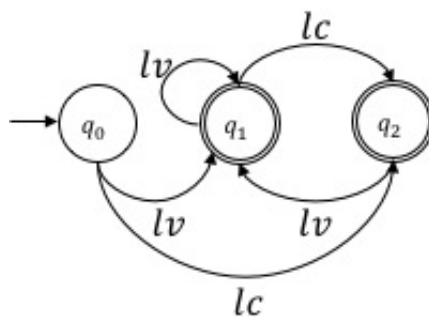


FIGURE 14 – Exercice 2-3 : Solution 2