# TD : Automates finis déterministes

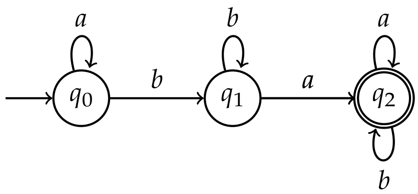
## Exercice 1

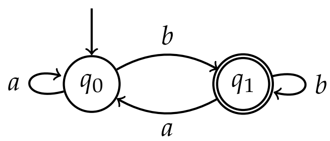
Soit $\Sigma = \{a, b\}$. Dans chaque cas, donner un automate fini déterministe reconnaissant le langage proposé.

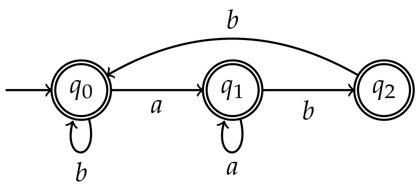
1. $L_1 = \varnothing$.
2. $L_2 = \{\varepsilon\}$.
3. $L_3 = \{x\}$ avec $x \in \Sigma$.
4. $L_4 = \{$mots commençant par $bab\}$.
5. $L_5 = \{$mots finissant par $bab\}$.
6. $L_6 = \{$mots contenant le facteur $bab\}$.
7. $L_7 = \{$mots ne contenant pas le facteur $bab\}$.

## Exercice 2

Décrire, en justifiant, le langage reconnu par l'automate proposé.

1. 


2. 


3. 


## Exercice 3

Soit $A = (Q, q_0, I, \delta)$ est un automate fini déterministe. On a défini en cours la fonction de transition étendue aux mots $\delta^* : Q \times \Sigma^* \to Q$ par induction :

$$\forall q \in Q,\ \delta^*(q, \varepsilon) = q$$

$$\forall q \in Q,\ \forall u \in \Sigma^*,\ \forall x \in \Sigma,\ \delta^*(q, xu) = \delta^*(\delta(q, x), u)$$

Démontrer les propositions suivantes :

$$\forall q \in Q,\ \forall u \in \Sigma^*,\ \forall x \in \Sigma,\ \delta^*(q, ux) = \delta(\delta^*(q, u), x)$$

$$\forall q \in Q,\ \forall u \in \Sigma^*,\ \forall v \in \Sigma^*,\\ \delta^*(q, uv) = \delta^*(\delta^*(q, u), v)$$

## Exercice 4

Vous arrivez dans une salle avec 3 interrupteurs lumineux qui peuvent être éteints ou allumés :

- Quand on appuie sur l'interrupteur de gauche il s'allume.
- Quand on appuie sur l'interrupteur du milieu, ceux de gauche et de droite changent d'état.
- Quand on appuie sur l'interrupteur de droite lorsqu'il est allumé, il s'éteint et les deux autres changent d'état. Sinon il ne se passe rien.

Au départ tous les interrupteurs sont éteints et vous devez tous les allumer.

1. Modéliser cette situation à l'aide d'un automate fini déterministe.
2. Déterminer une solution de longueur minimale.
3. Est-il possible de résoudre le jeu sans jamais appuyer sur l'interrupteur de gauche ? du milieu ? de droite ?

## Exercice 5

On considère le langage $L$ sur $\Sigma = \{a, b, c\}$ pour lequel :
- après l'apparition d'un premier $b$, on ne peut plus lire de $a$ dans la suite;
- après l'apparition d'un premier $c$, on ne peut plus lire de $b$ dans la suite.

1. Donner un automate $A_1$ reconnaissant le langage $L$.
2. Donner un automate $A_2$ reconnaissant le langage $\overline{L}$.

## Exercice 6

On considère un jeu à deux joueurs $A$ et $B$, qui se joue avec un paquet de cartes qui sont de 3 sortes :
- une carte $A$ fait gagner 0 point au joueur 1 et 1 point au joueur 2,
- une carte $B$ fait gagner 2 points au joueur 1 et 0 point au joueur 2,
- une carte $C$ fait gagner 1 point au joueur 1 et 2 points au joueur 2.

Si le score d'un des joueurs dépasse 2 points après avoir tiré une carte alors son score est ramené à 0. Une partie est un mot sur $\Sigma = \{A, B, C\}$ qui représente les cartes tirées.

1. Déterminer un automate $A_1$ qui accepte les parties pour lesquelles le joueur 1 termine avec 2 points.
2. Déterminer un automate $A_2$ qui accepte les parties pour lesquelles le joueur 2 termine avec 2 points.
3. Décrire à l'aide d'un automate les parties qui se terminent sur un score maximal (2-2) pour les deux joueurs.
4. Tous les scores sont-ils possibles ?
5. À l'aide de ce qui précède, déterminer un automate acceptant les parties qui conduisent au gain du joueur 1 sur le joueur 2.

## ⋆Exercice 7

**Entiers en représentation binaire** Dans cet exercice on considère $\Sigma = \{0, 1\}$. Un entier $n \in \mathbb{N}$ sera codé par un mot sur $\Sigma$, d'après sa représentation en base 2 avec le bit de poids faible à droite. Par exemple, 10 est représenté par le mot $u = 1010$.

1. Démontrer que le langage des entiers pairs est reconnaissable.
2. Démontrer que le langage des entiers multiples de 4 est reconnaissable.
3. Démontrer que le langage des entiers multiples de 5 est reconnaissable.
4. En déduire un automate qui accepte les entiers terminant par 0 en base 10.

## ⋆Exercice 8

**Exemples de langages non reconnaissables** Soit $\Sigma = \{a, b\}$. On considère le langage

$$L_1 = \{a^n b^n,\ n \in \mathbb{N}\}$$

et le langage

$$L_2 = \{\text{mots contenant autant de } a \text{ que de } b\}.$$

On suppose qu'il existe un automate fini déterministe complet $A_1$ à $N$ états $Q = \{q_0, \dots, q_{N-1}\}$ qui reconnait $L_1$. On considère l'application $\varphi : \mathbb{N} \to Q$ définie par $\varphi(k) = \delta^*(q_0, a^k)$.

1. Justifier que $\varphi$ n'est pas injective.
2. En déduire que le langage $L_1$ n'est pas reconnaissable par automate fini déterministe.
3. Montrer que le langage $L_2$ n'est pas reconnaissable par automate fini déterministe.

## ⋆Exercice 9

Déterminer un langage reconnaissable $L$ tel qu'il n'existe pas d'automate fini déterministe ayant un seul état final qui le reconnaît.