Exercice 1. Construction d'automates déterministes

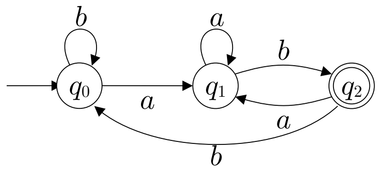
Pour chacun des langages suivants sur l'alphabet $\Sigma = \{a, b\}$, donner un automate fini déterministe complet reconnaissant L :

1. Le langage de tous les mots de longueur impaire.
2. Le langage de tous les mots contenant au plus une occurrence de la lettre a.
3. Le langage de tous les mots avec un nombre pair de a.
4. Le langage de tous les mots u tels que $|u|_a$ est pair et $|u|_b$ est un multiple de 3.
5. Le langage $L = ab\Sigma^* \cup ba\Sigma^*$.
6. Le langage de tous les mots comptant au moins deux occurrences de leur dernière lettre.

Exercice 2. Standardisation d'automates

Un automate fini déterministe est dit ***standard*** s'il ne contient pas de transition vers l'état initial.

1. Construire un automate standard reconnaissant le même langage que :



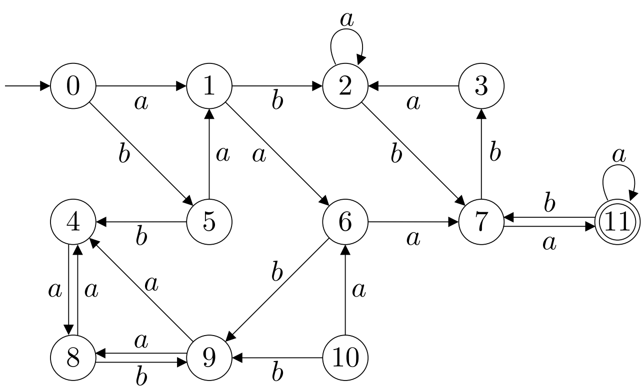
2. Soit L un langage reconnaissable. Construire un automate standard reconnaissant L.

Exercice 3.

Trouver un langage reconnaissable ne pouvant pas être reconnu par un automate fini déterministe avec un seul état acceptant. Justifier

Exercice 4. Émondage

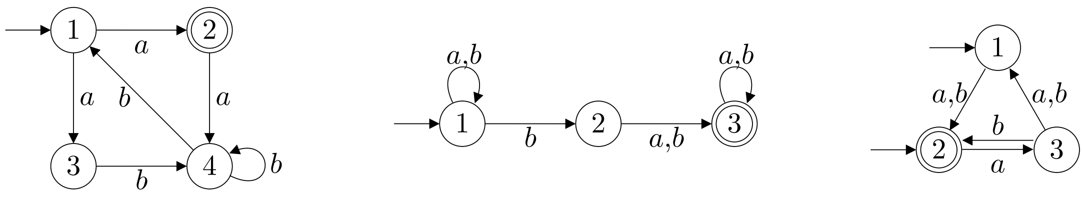
Émonder l'automate fini déterministe suivant :



Exercice 5. Automates non déterministes

1. (a) Construire un automate fini non déterministe qui reconnaît le langage $\{\epsilon, a, ba, aba, baba\}$.
   (b) Même question avec un automate fini déterministe.
2. (a) Construire un automate fini non déterministe qui reconnaît le langage $L = \{a\}\{a, b\}^*\{a\}$.
   (b) Construire un automate déterministe à partir de l'automate obtenu dans la question précédente.

3. Déterminiser les automates suivants :



Exercice 6. Langage singleton

1. Soit $A = (Q, I, F, \delta)$ un automate fini non déterministe tel que $\mathscr{L}(A) \neq \varnothing$. Montrer qu'il existe $u \in \mathscr{L}(A)$ tel que $|u| \leqslant \text{card}(Q) - 1$.
2. Soit $n \in \mathbb{N}^*$, $u \in \Sigma^n$ et $L = \{u\}$.
   (a) Montrer que L est reconnu par un automate fini déterministe ayant $n+1$ état.
   (b) Montrer qu'aucun automate fini non déterministe avec moins de n états ne peut reconnaître L.

Exercice 7. Écriture binaire

Soit $\Sigma = \{0, 1\}$. Pour tout mot $u = u_1 \ldots u_n$, on note $b_1(u)$ et $b_2(u)$ les entiers :

$$b_1(u) = \sum_{i=1}^{n} u_i 2^{n-i} \qquad\qquad b_2(u) = \sum_{i=1}^{n} u_i 2^{i-1}$$

En d'autres termes, u est l'écriture en base 2 de $b_1(u)$ (resp. $b_2(u)$) en commençant par le bit de poids fort (resp. de poids faible). Par exemple :

u	ϵ	0	10	01001	011001010
$b_1(u)$	0	0	2	9	202
$b_2(u)$	0	0	1	18	166

Pour tout $k \in \mathbb{N}^*$, on définit :

$$A_k = \Big\{u \in \Sigma^* : b_1(n) \text{ est divisible par } k\Big\} \qquad\qquad B_k = \Big\{u \in \Sigma^* : b_2(n) \text{ est divisible par } k\Big\}$$

1. Montrer que A_2, A_3, A_4, B_2, B_3, B_4 sont reconnaissables. Pour cela, on construira des automates finis déterministes avec un minimum d'états.
2. Montrer que le langage A_k est reconnaissable pour tout $k \in \mathbb{N}^*$. Justifier.
3. Montrer que le langage B_k est reconnaissable pour tout $k \in \mathbb{N}^*$. Justifier.

Exercice 8. Facteurs, sous-mots et préfixes

Pour chacun des langages suivants sur l'alphabet $\Sigma = \{a, b\}$, donner un automate fini déterministe reconnaissant L :

1. Le langage de tous les mots ayant aba pour sous-mot.
2. Le langage de tous les mots ayant aba pour facteur.

Pour tout $u \in \Sigma^*$, on note $\text{pref}(u)$ l'ensemble des préfixes de u et $A(u)$ l'automate fini déterministe (Q, q_0, F, δ) où :

$$Q = \text{pref}(u) \qquad\qquad q_0 = \epsilon \qquad\qquad F = \{u\}$$

et pour tout $v \in Q$ et $a \in \Sigma$, $\delta(v, a)$ est le plus long suffixe de va appartenant à Q.

3. Construire $A(abaca)$ dans le cas où $\Sigma = \{a, b, c\}$.
4. Soit $u \in \Sigma^*$. Déterminer $\mathscr{L}(A(u))$. Justifier.
5. Soit $u \in \Sigma^*$. Définir un automate fini déterministe qui reconnaît les mots dont u est un facteur.