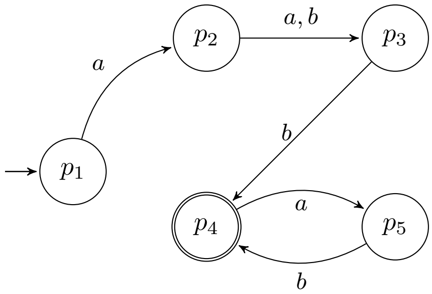
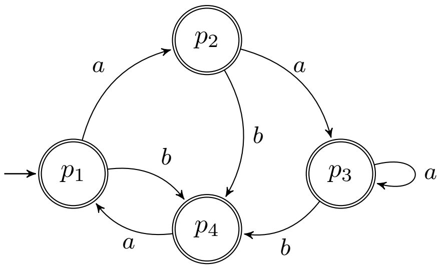
TD Théorie des langages 1 — Feuille 3

Langages réguliers – Déterminisation, Minimisation

Exercice 6 Minimiser les automates suivants :



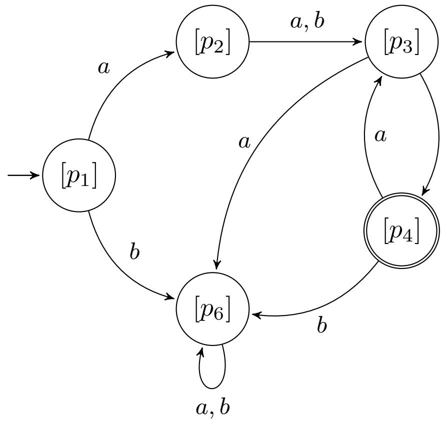


Solution de l'Exercice 6. Les automates sont déterministes mais non complets, il faut penser à ajouter un état puits.

Les premières classes (celles de $\equiv_0$) sont $Q \setminus F$ et F. Dans la suite, on détermine la classe $\equiv_{k+1}$ en fonction de la classe $\equiv_k$ et de la « signature » de chaque état, c.-à-d. des classes vers lesquelles on arrive après une transition par chacun des symboles du vocabulaire. Par exemple, la signature CD signifie qu'en lisant un a, on arrive dans la classe C et qu'en lisant un b, on arrive dans la classe D. On rappelle qu'il est inutile de traiter les classes singletons car elles ne peuvent pas diminuer d'avantage.

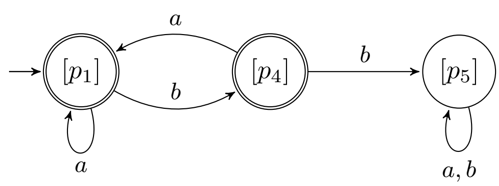
Premier automate : L'état p_6 est l'état puit qu'on a rajouté pour compléter l'automate.

$\equiv_0$	$\{p_1,$	$p_2,$	$p_3,$	p_5	$p_6\}$	$\{p_4\}$
noms des classes			A			B
« signatures »	AA	AA	AB	AB	AA	
$\equiv_1$	$\{p_1,$	$p_2,$	$p_6\}$	$\{p_3,$	$p_5\}$	$\{p_4\}$
noms des classes		C		D		B
« signatures »	CC	DD	CC	CB	CB	
$\equiv_2$	$\{p_1,$	$p_6\}$	$\{p_2\}$	$\{p_3,$	$p_5\}$	$\{p_4\}$
noms des classes	E		G	D		B
« signatures »	GE	EE		EB	EB	
$\equiv_3$	$\{p_1\}$	$\{p_6\}$	$\{p_2\}$	$\{p_3,$	$p_5\}$	$\{p_4\}$
noms des classes	H	I	G	D		B
« signatures »				IB	IB	
$\equiv_4$	$\equiv_3$					



Second automate : L'état p_5 est l'état puit, rajouté pour compléter l'automate.

$\equiv_0$	$\{p_5\}$	$\{p_1,$	$p_2,$	$p_3,$	$p_4\}$
noms des classes	A		B		
« signatures »		BB	BB	BB	BA
$\equiv_1$	$\{p_5\}$	$\{p_1,$	$p_2,$	$p_3\}$	$\{p_4\}$
noms des classes	A		C		D
« signatures »		CD	CD	CD	
$\equiv_2$	$\equiv_1$				

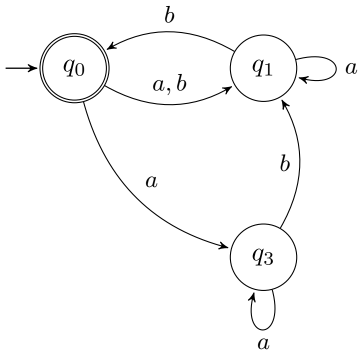


$$
\begin{aligned}
\equiv_0 &\;:\; \{p_1, p_2, p_3, p_4\},\ \{p_5\} \\
\equiv_1 &\;:\; \{p_1, p_2, p_3\},\ \{p_4\},\ \{p_5\} \\
\equiv_2 &\;:\; \equiv_1
\end{aligned}
$$

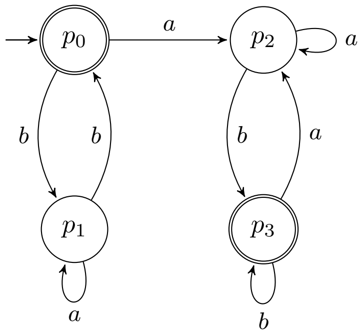
Exercice 7 Soit $Q = \{q_0, q_1, q_2, q_3, q_4\}$. Déterminiser et minimiser l'automate $A = (Q, \{a, b\}, \delta, \{q_0\}, \{q_4\})$, où la relation de transition δ est récapitulée ci-dessous :

δ	a	b	ε
q_0	q_1	$\times$	q_3, q_4
q_1	q_1	q_0	$\times$
q_2	$\times$	q_4	q_1
q_3	q_3	q_1	$\times$
q_4	$\times$	$\times$	q_3

Solution de l'Exercice 7. L'état q_2 n'est pas accessible. On le retire donc, et après élimination des ε-transitions, on a :



Après déterminisation, en posant $p_0 = \{q_0\}$, $p_1 = \{q_1\}$, $p_2 = \{q_1, q_3\}$ et $p_3 = \{q_0, q_1\}$, on obtient :



Minimisation :

$$\begin{array}{lcl} \equiv_0 & : & \{p_0, p_3\},\ \{p_1, p_2\} \\ \equiv_1 & : & \{p_0\},\ \{p_3\},\ \{p_1, p_2\} \\ \equiv_2 & : & \{p_0\},\ \{p_3\},\ \{p_1\},\ \{p_2\} \\ \equiv_3 & : & \equiv_2 \end{array}$$

Exercice 9 [Avancé] Pour $k > 0$, soit L_k le langage constitué des mots sur $\{0, 1\}$ de longueur au moins k, et dont le $k^{\text{ième}}$ symbole **en partant de la fin** est un 1. Par exemple, 00101 et 100110111 sont dans L_3. Formellement,

$$L_k \stackrel{\text{def}}{=} \{a_1 \ldots a_n \mid n \geq k \wedge a_{n-k+1} = 1\}.$$

▷ Question 1 Construire un automate (non-déterministe) à $k+1$ états qui reconnaît L_k.

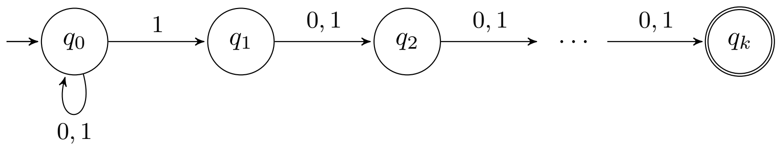
▷ Question 2 Construire un automate déterministe complet minimal reconnaissant L_2.

On cherche à borner la taille minimale d'un automate déterministe complet reconnaissant L_k. Soit $A = (Q, \{0, 1\}, \delta, \{q_0\}, F)$ un automate déterministe complet reconnaissant L_k. On définit $f : \{0, 1\}^k \to Q$, qui à tout mot u de longueur k associe $\delta^*(q_0, u)$. Autrement dit, $f(u)$ est l'état atteint par le chemin de trace u dans A, partant de q_0.

▷ Question 3 Montrer que f est injective. En déduire une borne inférieure de la taille de A.

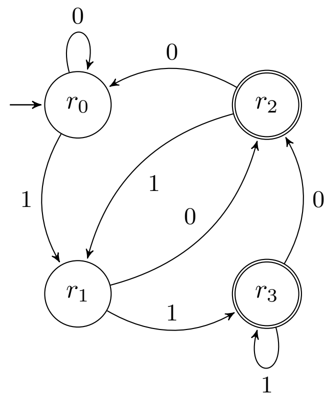
Solution de l'Exercice 9.

▷ Question 1 Automate non-déterministe :



▷ Question 2 Table de transition de l'automate déterministe :

I/F	nom	δ	0	1
I	r_0	q_0	q_0	q_0, q_1
	r_1	q_0, q_1	q_0, q_2	q_0, q_1, q_2
F	r_2	q_0, q_2	q_0	q_0, q_1
F	r_3	q_0, q_1, q_2	q_0, q_2	q_0, q_1, q_2



▷ Question 3 Supposons que f n'est pas injective. Il existe donc deux mots u et v de longueur k tels que $u \neq v$ et $f(u) = f(v)$. Notons en particulier que ceci signifie que pour tout $x \in \{0,1\}^*$, on a $\delta^*(q_0, ux) = \delta^*(\delta^*(q_0, u), x) = \delta^*(\delta^*(q_0, v), x) = \delta^*(q_0, vx)$.

Comme u et v sont distincts, leur plus grand préfixe commun w est un préfixe strict. Sans perte de généralité, on suppose que $u = w1u_1$ et $v = w0v_1$ (noter que $|u_1| = |v_1|$). Posons $x = 0^{|w|}$.

Soient $u' = ux$ et $v' = vx$. Alors $|1u_1x| = k$ (on a enlevé w au début et ajouté x à la fin par rapport à u), ce qui prouve que $u' = w1u_1x$ est élément de L_k ; donc $\delta^*(q_0, u') \in F$. Mais de la même façon, on vérifie que $v' \notin L_k$ et donc $\delta^*(q_0, v') \notin F$. Or, $\delta^*(q_0, u') = \delta^*(q_0, ux) = \delta^*(q_0, vx) = \delta^*(q_0, v')$; on a donc une contradiction. On en déduit que f est bien injective.

La fonction f étant injective, on en déduit que $Q \geq 2^k$ (car le cardinal de $\{0,1\}^k$ est 2^k).

Une façon intuitive de comprendre ce résultat est la suivante : pour accepter un mot de L_k, un automate déterministe doit savoir si le k^{e} symbole en partant de la fin est un 1 ou non. Après une transition, le nouveau k^{e} symbole en partant de la fin était avant cette transition le $(k-1)^{\text{e}}$ symbole en partant de la fin. Ainsi, il faut non seulement se souvenir si le k^{e} symbole était un 1, mais également si le $(k-1)^{\text{e}}$ l'était (toujours en partant de la fin) pour maintenir cette information après une transition. Par le même raisonnement, il faut savoir si chacun des k derniers symboles sont ou non des 1. Cette information coûte un bit par symbole, il nous faut donc 2^k états pour reconnaître L_k.