Licence Sciences et Technologies
Univ. Grenoble Alpes

Examen à mi-parcours
INF 302 : Langages et Automates
L2, 2016/2017

Rappel des consignes et quelques conseils/remarques

— Durée : 2 heures.
— Aucune sortie avant 30 minutes. Aucune entrée après 30 minutes.
— Tout document du cours ou du TD est autorisé.
— Tout dispositif électronique est interdit (calculatrice, téléphone, tablette, etc.).
— Le soin de la copie sera pris en compte (**-1 point en cas de manque de soin**).
— Les exercices sont indépendants.
— Le barème est donné à titre indicatif. L'examen est sur 22 points.

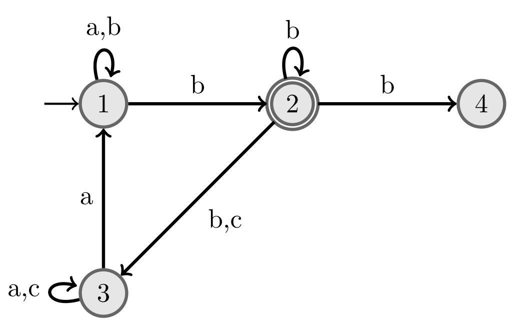
Exercice 1 (Vrai ou Faux - 2 points)

Répondre par Vrai ou Faux aux questions suivantes. Justifier *soigneusement* vos réponses (max 3 lignes).

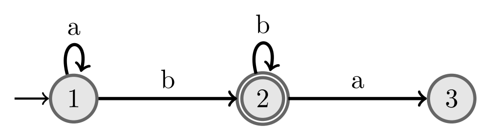
1. Étant donné un langage, on peut trouver un automate d'états finis déterministe qui reconnaît ce langage.
2. Étant donné un langage régulier, on peut trouver un automate d'états finis déterministe (sans ϵ-transition) qui reconnaît ce langage.
3. Si L est d'états finis, alors pour tout langage $L' \supseteq L$, L' est d'états finis.
4. Soit L un langage régulier et n une constante d'itération pour L. Alors, $2 \times n$ est également une constante d'itération pour L.

Solution de l'exercice 1

1. Faux. Pour les langages qui ne sont pas d'états finis, il n'est pas possible de trouver d'automate d'états finis déterministe. Par exemple, nous pouvons considérer le langage des mots qui contiennent autant de a que de b.
2. Vrai. Par le théorème de Kleene, les langages réguliers sont les langages d'états finis. Chaque expression régulière peut être traduite en automate non-déterministe avec ϵ-transitions. Chaque automate non-déterministe avec ϵ-transitions peut être déterminisé.
3. Faux. Par exemple, le langage vide est d'états-finis. Le langage des mots qui contiennent autant de a que de b n'est pas d'états finis et contient le langage vide.
4. Vrai. On sait que si n est une constante d'itération d'un langage régulier, alors tout entier $n' \geq n$ est également une constante d'itération pour ce langage.

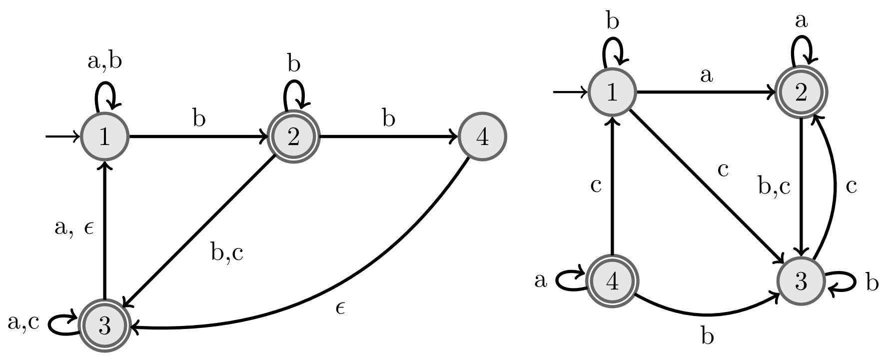


(a) Un automate pour l'exercice 3



(b) Un automate pour l'exercice 4

FIGURE 1: Des automates



(a) Un automate pour l'exercice 7 (b) Un automate pour l'exercice 8

Figure 2: Des automates pour les exercices 7 et 8

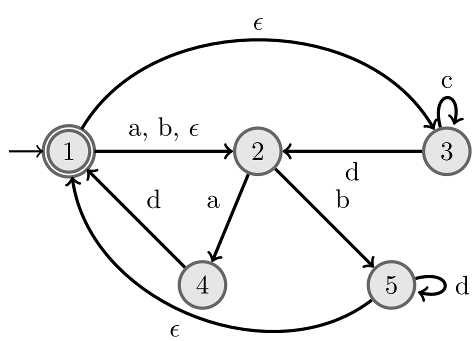
Exercice 2 (Expression régulière vers automate - 2 points)

1. Donner un automate non-déterministe avec ϵ-transitions qui reconnait le langage dénoté par l'expression régulière

$$((a + b + c^* \cdot d + \epsilon) \cdot (a \cdot d + b \cdot d^*) + \epsilon)^* .$$

Solution de l'exercice 2

1. Un automate reconnaissant l'expression régulière est donné ci-dessous :



Exercice 3 (Automate vers expression régulière - 3 points)

Nous considérons l'automate représenté dans la Figure 1a.

1. Donner une expression régulière dénotant le langage reconnu par cet automate en utilisant la méthode associant des équations aux états.

Solution de l'exercice 3

1. Le système d'équations associé à cet automate est :

$$
\begin{aligned}
X_1 &= (a + b)X_1 + bX_2 \\
X_2 &= bX_2 + (b + c)X_3 + bX_4 + \epsilon \\
X_3 &= (a + c)X_3 + aX_1 \\
X_4 &= \emptyset
\end{aligned}
$$

Nous pouvons simplifier l'équation associée à X_2 en $X_2 = bX_2 + (b+c)X_3 + \epsilon$. En utilisant le lemme d'Arden sur l'équation associée à X_2, comme $\epsilon \notin b$, nous avons $X_2 = b^*((b+c)X_3 + \epsilon)$. En utilisant le lemme d'Arden sur l'équation associée à X_3, comme $\epsilon \notin a+c$, nous avons $X_3 = (a+c)^*aX_1$. Nous pouvons injecter X_3 dans X_2, ce qui donne $X_2 = b^*((b+c)(a+c)^*aX_1 + \epsilon)$. Nous pouvons injecter X_2 dans l'équation associée à X_1, ce qui donne $X_1 = (a+b)X_1 + b^+((b+c)(a+c)^*aX_1 + \epsilon) = (a+b+b^+(b+c)(a+c)^*a)X_1 + b^+$. En appliquant le lemme d'Arden sur l'équation associée à X_1, comme $\epsilon \notin (a+b+b^+(b+c)(a+c)^*a)$, nous obtenons : $(a+b+b^+(b+c)(a+c)^*a)^*b^+$.

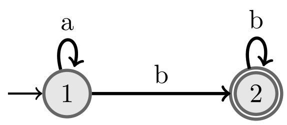
Exercice 4 (Automate vers expression régulière - 3 points)

Nous considérons l'automate représenté dans la Figure 1b.

1. Donner une expression régulière dénotant le langage reconnu par cet automate en utilisant la méthode associant des équations aux chemins.

Solution de l'exercice 4

1. Comme l'état 3 n'est pas co-accessible, l'automate suivant est équivalent à l'automate donné en énoncé.



Nous calculons les expressions régulières $R^k_{i,j}$ pour $k = 0$ jusqu'à $k = 2$. Rappelons que pour $k \geq 1$, $R^k_{i,j} = R^{k-1}_{i,j} + R^{k-1}_{i,k}(R^{k-1}_{k,k})^* R^{k-1}_{k,j}$.

$$
\begin{array}{lll}
R^0_{1,1} = a + \epsilon & R^1_{1,1} = a^* & R^2_{1,1} = a^* \\
R^0_{1,2} = b & R^1_{1,2} = a^*b & R^2_{1,2} = a^*b^+ \\
R^0_{2,1} = \emptyset & R^1_{2,1} = \emptyset & R^2_{2,1} = \emptyset \\
R^0_{2,2} = b + \epsilon & R^1_{2,2} = b + \epsilon & R^2_{2,2} = b^*
\end{array}
$$

L'équation régulière associée à cet automate est $R^2_{1,2} = a^*b^+$.

Exercice 5 (Automate vers expression régulière, borne supérieure - 3 points)

Soit $A = (Q, q_0, \Sigma, \delta, F)$ un automate d'états finis déterministe dont on cherche à calculer une expression régulière en utilisant la méthode associant les équations aux chemins. Nous supposons que Q est un interval de $\mathbb{N}$ dont la borne inférieure est 1. Rappelons que pour $i, j \in Q$ et $0 \leq k \leq |Q|$, $R^k_{i,j}$ est l'expression régulière dénotant les chemins partant de l'état i et allant jusqu'à l'état j et passant par des états intermédiaires $q \leq k$.

1. Rappeler l'expression régulière dénotant le langage reconnu par A.
2. Donner une borne supérieure au nombre d'équations qu'il faut calculer pour trouver l'expression régulière dénotant le langage reconnu par l'automate en fonction de $|Q|$.
3. Donner une borne supérieure au nombre d'équations qu'il faut calculer pour trouver l'expression régulière dénotant le langage reconnu par l'automate en fonction de $|Q|$ et $|F|$.

Solution de l'exercice 5

1. Supposons que l'état initial est l'état 1. L'expression régulière est :

$$\sum_{f \in F} R^{|Q|}_{1,f}.$$

2. Il faut calculer les $R^k_{i,j}$ pour k allant de 0 à $|Q|$, c'est-à-dire $|Q|+1$ étapes. A chaque étape, pour un k donné, il y a $|Q| \times |Q|$ expressions régulières $R_{i,j}$ à calculer. Au final, cela fait $(|Q|+1) \times |Q| \times |Q|$ expressions régulières à calculer.

3. Nous observons que lors de la dernière étape, il n'est pas nécessaire de calculer tous les $R_{i,j}^{|Q|}$ mais uniquement les $R_{1,f}^{|Q|}$ pour $f \in F$. On obtient $|Q|^3 + |F|$ expressions régulières à calculer.

Exercice 6 (Constante d'itération minimale - 2 points)

1. Donner la constante d'itération minimale du langage dénoté par l'expression régulière $(a \cdot b)^*$.

Solution de l'exercice 6

1. La constante d'itération minimale est 2. En effet, le mot ϵ est dans le langage mais ne peut pas être itéré. De plus, il n'y a pas de mot de longueur 1 qui soit dans le langage. Soit w un mot dans le langage tel que $|w| \geq 2$. Alors w est de longueur paire et peut s'écrire $w = w_0 \cdots w_n$ avec $n \in \mathbb{N}$ et $\forall i \in [0, n] : w_i = a \cdot b$. Le mot w peut être itéré à travers la décomposition $x = \epsilon$, $y = w_0$ et $z = w_1 \cdots w_n$ si $n \geq 1$ et $z = \epsilon$ sinon.

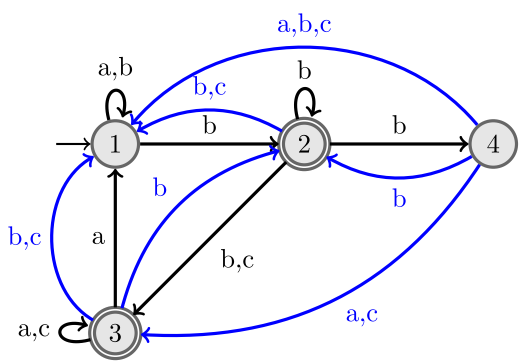
Exercice 7 (Transformations d'automate - 4 points)

Nous considérons l'automate d'états finis non déterministe avec ϵ-transitions représenté dans la Figure 2a.

1. Supprimer les ϵ-transitions, c'est-à-dire, donner un automate d'états finis non-déterministe sans ϵ-transitions qui reconnaît le même langage.
2. Déterminiser l'automate obtenu à la question précédente.

Solution de l'exercice 7

1. L'automate ci-dessous est l'automate résultant de la suppression des ϵ-transitions.



2. Nous représentons l'automate de la question précédente sous forme tabulaire avant déterminisation.

	1	2	3	4
a	1		1, 3	1, 3
b	1, 2	1, 2, 3, 4	1, 2	1, 2
c		1, 3	1, 3	1, 3

Nous appliquons l'algorithme de déterminisation. Nous obtenons l'automate représenté par le tableau suivant.

	1	1, 2	1, 3	1, 2, 3, 4
a	1	1	1, 3	1, 3
b	1, 2	1, 2, 3, 4	1, 2	1, 2, 3, 4
c		1, 3	1, 3	1, 3

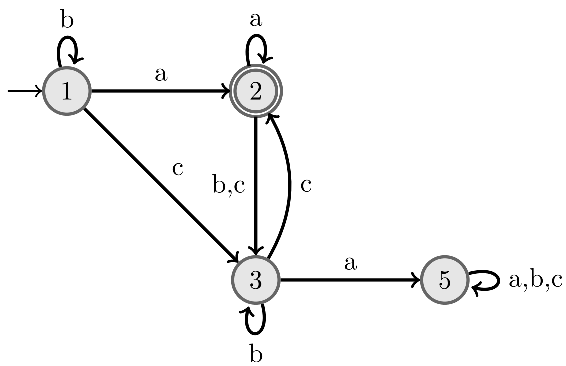
Exercice 8 (Minimisation d'automate - 3 points)

Nous considérons l'automate d'états finis déterministe représenté dans la Figure 2b.

1. Donner l'automate minimisé reconnaissant le langage reconnu par cet automate.

Solution de l'exercice 8

1. Nous appliquons l'algorithme de minimisation sur l'automate donné en énoncé. L'alphabet de l'automate est $\{a, b, c\}$. Nous complétons d'abord l'automate car il n'y a pas de transition définie à partir de l'état 3 sur le symbole a. De plus, nous supprimons l'état 4 et les transitions qui en sortent car cet état n'est pas accessible.



Les étapes du calcul sont représentées ci-dessous.

$\equiv_0$	$\equiv_1$
2	2
1	1
3	3
5	5

Chaque classe d'équivalence est de cardinal 1. L'automate est donc minimal.