

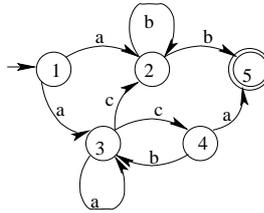
A.1 Automates et algorithmes

1. Soit $A = \{a, b, c\}$. Donner des automates déterministes complets reconnaissant les langages suivants :
 - (a) L'ensemble des mots dont la longueur est un multiple de 3.
 - (b) L'ensemble des mots dans lesquels le motif ab , s'il apparaît, est suivi de ccc .
 - (c) L'ensemble des mots se terminant par b .
 - (d) L'ensemble des mots ne se terminant pas par b .
 - (e) L'ensemble des mots non vides ne se terminant pas par b .
 - (f) L'ensemble des mots contenant exactement un b .
 - (g) L'ensemble des mots ne contenant aucun b .
 - (h) L'ensemble des mots contenant au moins un a et dont la première occurrence de a n'est pas suivie par un c .
 - (i) L'ensemble des mots comportant au moins 3 lettres et dont la troisième lettre à partir de la fin est un a ou un c .
2. Proposer un automate et une expression rationnelle pour le langage de tous les mots de $\{a, b, c\}^*$ dont cac est un sous-mot¹.
3. Soit $X = \{a, b, c, \dots, z\}$. Proposer un automate **déterministe** et **minimal** qui reconnaisse le langage $X^*issime^2$. Peut-on proposer une généralisation sur le nombre minimal d'états d'un automate reconnaissant X^*u pour $u \in X^*$, en fonction de la longueur de u ?
4. Donner un automate qui reconnaît le langage décrit par les expressions rationnelles suivantes, en appliquant la méthode vue en cours. Pour le troisième automate, on donnera aussi (sans nécessairement appliquer l'algorithme de suppression des ε -transitions) un automate le plus simple possible sans ε -transition.
 - $(a|c)(b|\varepsilon)d^*$
 - $(ab|ba)^*$
 - $(a|b)(a|b)^*$
5. Soit $A = \{a, b, c\}$. Donner des automates déterministes complets reconnaissant les langages suivants :
 - (a) L'ensemble des mots de longueur paire.
 - (b) L'ensemble des mots où le nombre d'occurrences de b est divisible par 3.
 - (c) L'ensemble des mots se terminant par b .
 - (d) L'ensemble des mots ne se terminant pas par b .
 - (e) L'ensemble des mots non vides ne se terminant pas par b .
 - (f) L'ensemble des mots contenant au moins un b .
 - (g) L'ensemble des mots contenant au plus un b .
 - (h) L'ensemble des mots contenant exactement un b .
 - (i) L'ensemble des mots ne contenant aucun b .
 - (j) L'ensemble des mots contenant au moins un a et dont la première occurrence de a n'est pas suivie par un c .
 - (k) L'ensemble des mots comportant au moins 3 lettres et dont la troisième lettre à partir de la fin est un a ou un c .

¹Un *sous-mot* de u est une sous-suite de lettres — non nécessairement contiguë — de u . À distinguer d'un *facteur*. Exemple : *pis* est un sous-mot de *produits*.

²Mots formés d'un mot quelconque de X^* suivi des lettres i, s, s, i, m, e .

6. Déterminer l'automate suivant (on demande juste un automate déterministe reconnaissant le même langage) :



« Vérifier » que l'automate déterminisé reconnaît le même langage en testant 3 mots appartenant au langage.

7. Soit l'expression rationnelle $(aa|b)^*(ca^*|ba^*b)$.
- Proposer un automate qui reconnaît le langage décrit par cette expression.
 - À partir de l'automate, proposer une grammaire régulière engendrant le même langage.
 - Donner un arbre syntaxique avec la grammaire précédente pour le mot $aabaab$.
8. Soit l'expression rationnelle $(a|bc)^*z(z|ba|ca^*)$.
- Proposer un automate reconnaissant le même langage, en appliquant l'algorithme vu en cours
 - Éliminer les ε -transitions
 - Déterminiser l'automate résultant
 - Minimiser l'automate résultant
9. Soit l'automate suivant :

	a	b	c	ε
\rightarrow 1	1,2	3	5	5
2	3	2	1	
\leftarrow 3			5	4,6
4	3		6	2
5	5	4,6	6	
\leftarrow 6				

- Proposer une grammaire régulière qui engendre le même langage.
 - Proposer un automate sans ε -transition qui reconnaît le même langage.
10. Soit la grammaire presque régulière suivante.
- $$\begin{aligned}
 S &\rightarrow aS \mid aB \mid \varepsilon \\
 A &\rightarrow cS \mid bB \\
 B &\rightarrow aA \mid bB \mid A \\
 C &\rightarrow cD \mid A \\
 D &\rightarrow cC \mid \varepsilon
 \end{aligned}$$
- Proposer un automate qui reconnaît le même langage.
 - En procédant éventuellement par étapes, proposer un automate **déterministe** le plus simple possible qui reconnaisse le même langage.
 - En déduire une version simplifiée de la grammaire initiale.