Mathématiques pour l'informatique
TD
Automates finis

Construction d'automates

Exercice 1

Construire un automate fini dont les états correspondent aux situations de famille possibles d'une personne (célibataire, marié, divorcé, veuf) et dont les flèches correspondent aux changements de situation possible. Etiqueter ces flèches par m (mariage), d (divorce) et v (veuvage). Les états finaux sont ceux où on peut mourir, et selon le code civil français on ne peut pas redevenir célibataire une fois on a été marié.
Votre automate est-il déterministe ?

Exercice 2

(a) Concevoir un automate reconnaissant un seul mot : « coucou ». Votre automate est-il déterministe ?

(b) Concevoir un automate reconnaissant le mot « coucou » mais aussi n'importe quel nombre de « cou » qui se suivent : « cou », « coucou », « coucoucou »… Votre automate est-il déterministe ?

Exercice 3

Construire des automates finis qui reconnaissent les langages suivants, et caractériser chaque automate comme déterministe ou non déterministe :

(a) L'ensemble des mots sur l'alphabet {0, 1} dont le dernier symbole est 0.

(b) L'ensemble des mots sur l'alphabet {0, 1} qui commencent et finissent par 1 ; construire un automate qui reconnaît ces mots et qui n'a qu'un seul état terminal. Prendre en considération le fait que le mot « 1 » vérifie lui aussi la condition.

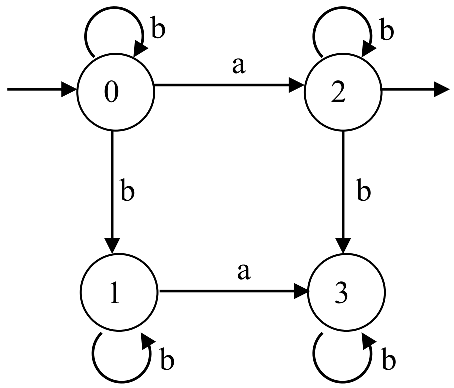
(c) L'ensemble des mots sur l'alphabet {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, –, .} qui représentent en C les constantes numériques. Une constante numérique en C peut avoir une des structures suivantes :
- Elle doit contenir au moins un chiffre ;
- Elle peut contenir un signe (+ ou –) (un seul) mais uniquement au début ;
- Elle peut contenir un point (.) (un seul) qui, s'il est présent, doit être précédé et/ou suivi par au moins un chiffre.

(d) L'ensemble des mots sur l'alphabet {0, 1} qui comportent au moins une fois le motif 10 et au moins une fois le motif 01 (on considère que, par exemple, la séquence 010 vérifie cette condition).

(e) $\{a^n b^m \mid n, m \in \mathbf{N}, n + m \text{ impair}\}$ (cela veut dire: s'il y a des *a*, ils précèdent à tous les *b* s'il en y a ; le nombre total de caractère est impaire).

Exercice 4
Construire un automate fini reconnaissant les entiers écrits en base 2 divisibles par 7.

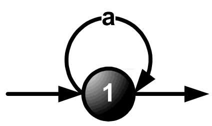
Exercice 5
Quel est le langage reconnu par l'automate suivant ? Quels états de cet automate sont inutiles, dans le sens que le langage reconnu par l'automate est le même qu'ils soit présents ou non ?



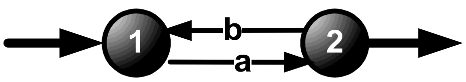
Standardisation

Exercice 6
Pour chacun des automates suivantes, (a) dire s'il est standard, (b) s'il ne l'est pas, le standardiser, (c) s'il reconnaît le mot vide, construire l'automate qui reconnaît le même langage à l'exception du mot vide :

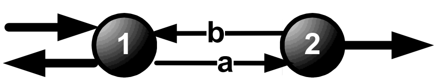
6-1



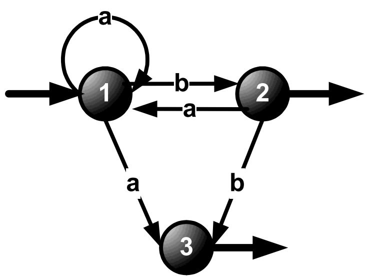
6.2



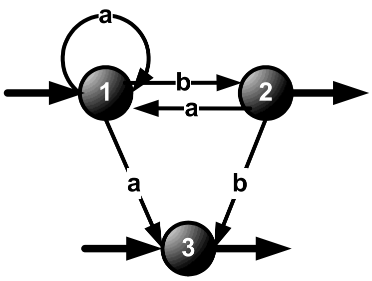
6.3



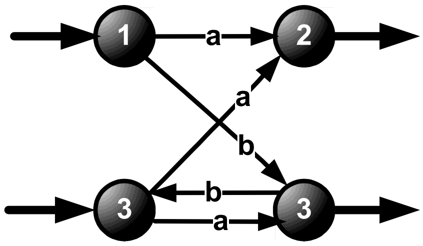
6.4



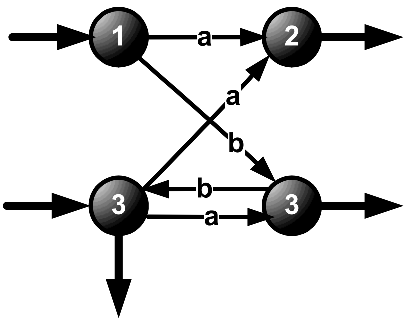
6.5



6.6



6.7



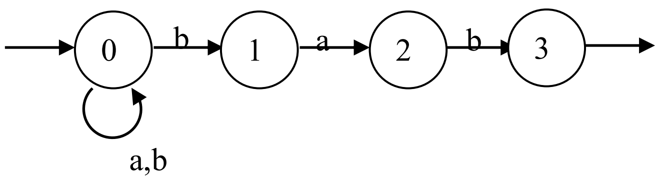
Déterminisation

Exercice 7

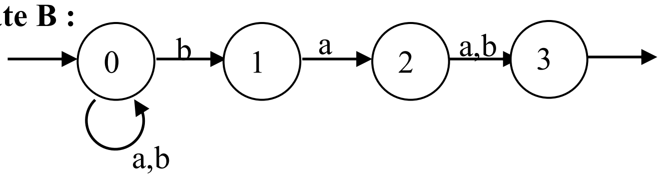
Voici cinq automates non déterministes notés A, B, C1, C2, C3, D. L'alphabet est $\sum$={a,b}.
Pour chacun de ces automates :

- Décrire le langage qu'il reconnaît (vous ne connaissez pas encore comment écrire des expressions rationnelles ; faites la description donc en langage ordinaire).
- Calculer l'automate déterministe équivalent et le compléter si besoin est.

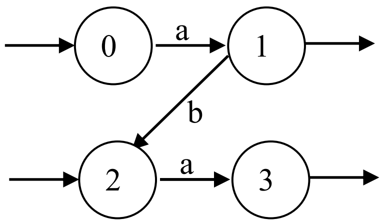
Automate A :



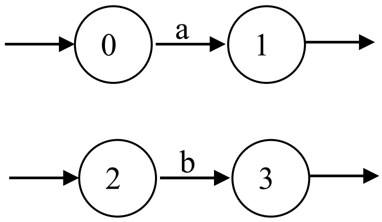
Automate B :



Automate C1:

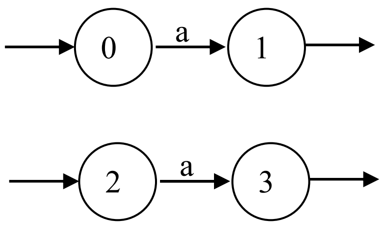


Automate C2:



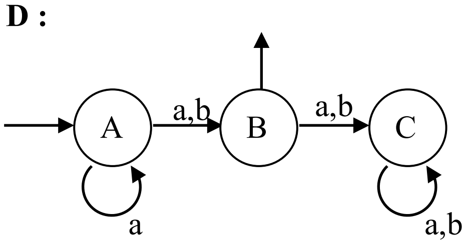
(Cet automate consiste en deux parties non connectées)

Automate C3:

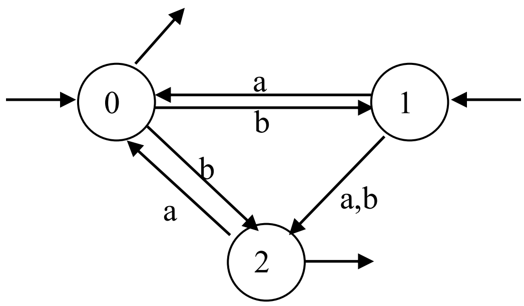


(Cet automate consiste en deux parties non connectées)

Automate D :



Exercice 8
Détermininiser et compléter l'automate suivant :
(Etats initiaux: 0 et 1 ; états terminaux : 0 et 2)



Langage complémentaire

Exercice 9
Construire des automates déterministes qui reconnaissent les langages suivants. En déduire des automates déterministes qui reconnaissent les complémentaires de ces langages :

(a) L'ensemble des mots sur l'alphabet {0, 1} qui contiennent exactement quatre fois le symbole 0.

(b) L'ensemble des mots sur l'alphabet {0, 1} qui contiennent au moins un 1.
(c) Sur l'alphabet {0, 1} : le mot vide et les mots '0', '1' et '01' .

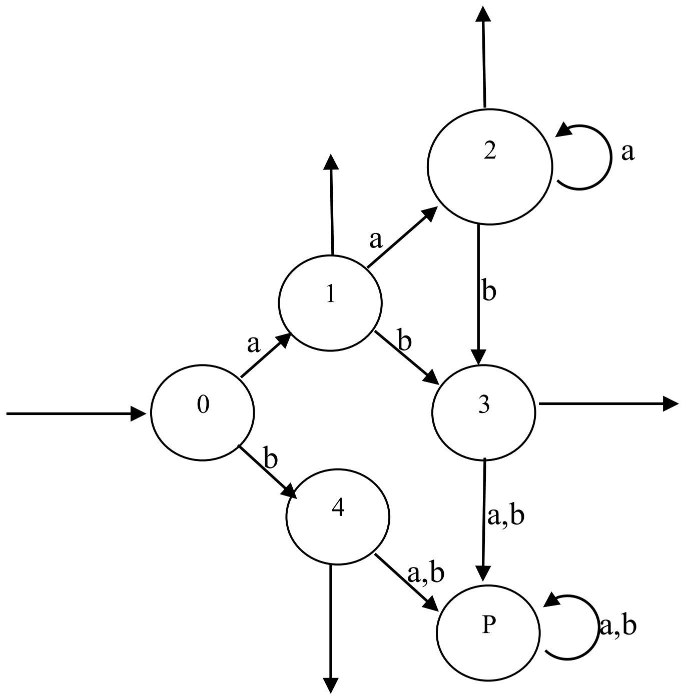
Langage complémentaire + minimisation

Exercice 10
Construire un automate fini reconnaissant l'ensemble des mots sur l'alphabet $A = \{a,b\}$ qui **ne se terminent pas** par *bbaa*. Le déterminiser et minimiser.

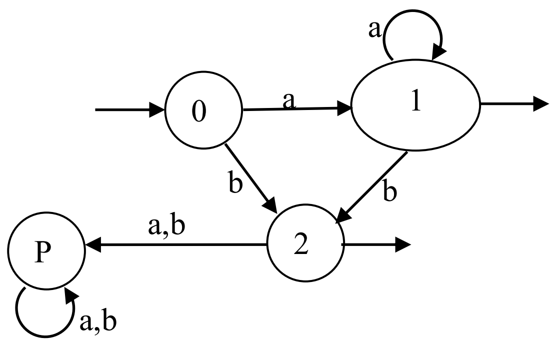
Exercice 11
Minimiser les deux automates déterministes équivalents obtenus en l'Exo 5 « D » (ici, nous avons renommé les états pour plus de simplicité) :

Le « grand » automate

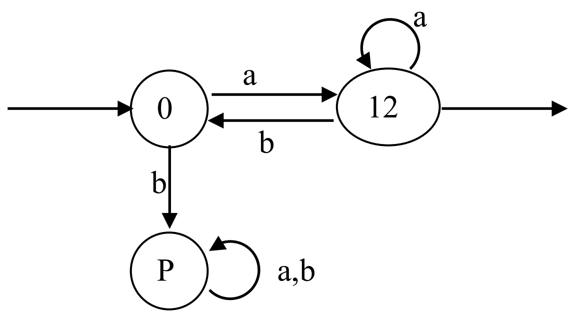
Etat	**a**	**b**
0	1	4
1	2	3
2	2	3
4	P	P
3	P	P
P	P	P



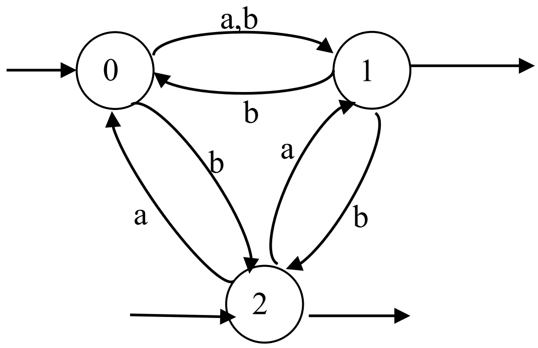
Et le « petit » automate



Exercice 12
Minimiser l'automate suivant :



Exercice 13
Déterminiser et minimiser l'automate suivant :



Exercice supplémentaire

On définit la famille d'automates suivants :

$$\tilde{A}_n = (Q_n, I, T, E),\ n \geq 1$$

avec

- $A = \{a, b\}$
- $Q_n = \{0, 1, \ldots, n\text{-}1\}$
- $I = T = \{0\}$
- Comme flèches étiquetées par ***a*** l'ensemble de (q.a.((q+1)**mod** n)) pour $\forall q : 0 \leq q \leq n-1$
- Comme flèches étiquetées par ***b*** l'ensemble de (q.b.0) et (q.b.q) pour $\forall q : 1 \leq q \leq n-1$ *(attention : la première inégalité commence par 0, et la seconde, par 1)*

Dessiner $\tilde{A}_3$ et $\tilde{A}_4$.

Puis montrer que le déterminisé complet de $\tilde{A}_n$ a toujours 2^n états.