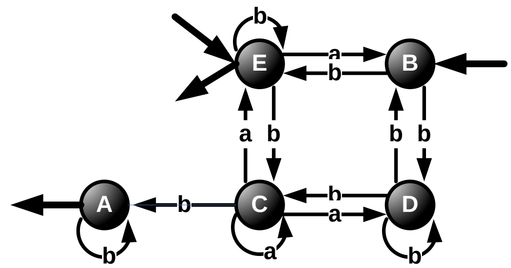
7DE: Mathématiques pour l'informatique

solutions

Exercice 1 Déterminisation.

Soit l'automate A_1 sur l'alphabet {a,b} :

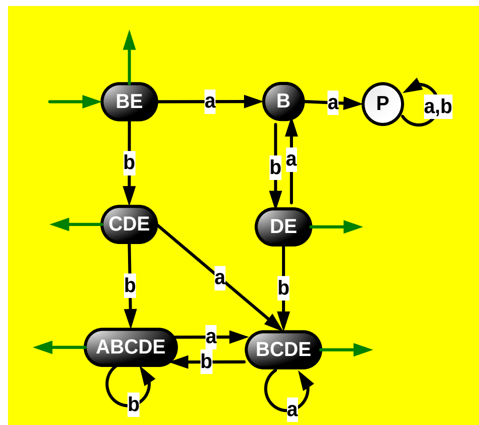
	état	a	b
S	A	-	A
E	B	-	D,E
	C	C,D,E	A
	D	-	B,C,D
E/S	E	B	C,E



Construire un automate déterministe complet A_{1dc} équivalent à cet automate.

Solution

	état	a	b
E/S	BE	B	CDE
	B	P	DE
S	CDE	BCDE	ABCDE
S	DE	B	BCDE
S	BCDE	BCDE	ABCDE
S	ABCDE	BCDE	ABCDE
	P	P	P



Exercice 2 Standardisation

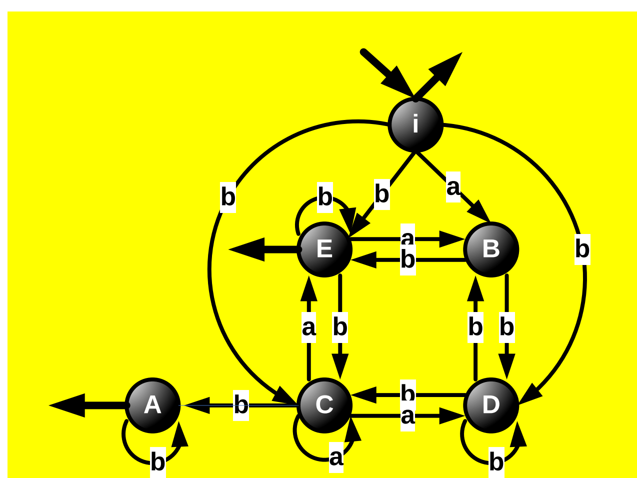
Obtenir un automate A_2 reconnaissant tout le langage reconnu par l'automate A_1 défini dans l'exercice 1.

Solution Il y a évidemment une erreur de copier-coller dans cette formulation, car si l'on la prend à la lettre, le l'automate le plus évident reconnaissant tout le langage reconnu par l'automate A_1 c'est tout simplement A_1 lui-même. Du fait que l'exercice est noté « standardisation », on peut deviner (?) que la formulation souhaitée était soit (a) « automate **standard** A_2 reconnaissant tout le langage reconnu par l'automate A_1 », soit, même plus probablement, (b) « automate A_2 reconnaissant tout le langage reconnu par l'automate A_1 à l'exception du mot vide ». Les deux interprétations sont acceptées.

a) Interprétation (a1) : on peut prendre l'automate d'origine tel quel et le standardiser, ou bien on peut obtenir un automate standard à partir de l'automate déterministe obtenu en premier exercice. Le premier choix donne l'automate suivant :

Détail :
l'entrée *i* est sortie car l'automate reconnait le mot vide, et les transitins de *i* sont obtenues ainsi :

EbE → ibE
EaB → iaB
EbC → ibC
BbE → ibE (une seconde fois)
BbD → ibD

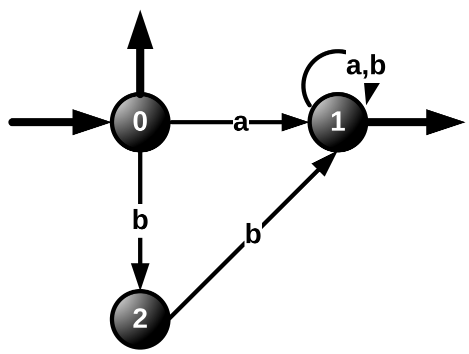


Le deuxième choix résulte en le même automate déterministe A_{1dc} qu'on avait obtenu car on remarque qu'il est standard. Le rôle de l'état *i* est joué par l'état BE.

b) Interprétation (b) : on enlève la flèche de sortie sur *i* ou sur BE.

Exercice 3 Langage complémentaire.

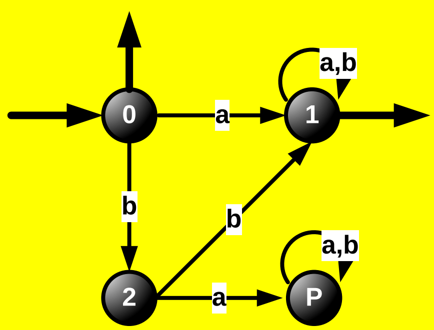
Soit l'automate A_3 suivant :



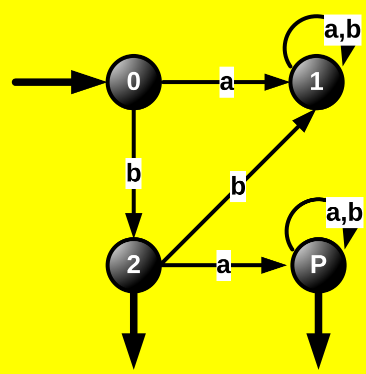
Construire un automate A_4 reconnaissant le langage complémentaire à celui reconnu par l'automate A_3.

Solution : L'automate A_3 est déterministe mais pas complet. Avant de le complémentariser, il faut le compléter. Voici le résultat :

Automate déterministe complet équivalent à A_3



Automate A_4 reconnaissant le langage complémentaire :



On voit que dans l'automate A_4, c'est l'état 1 qui est devenu une poubelle.

Exercice 4 Minimisation et le langage complémentaire

A = {a,b} est l'alphabet.

Pour l'automate A_5 défini par la table de transitions ci-dessous :

	état	a	b
S	0	--	--
S	1	3	5
E/S	2	4	0
S	3	0	1
S	4	3	5
S	5	4	0

a) Construire un automate <u>déterministe complet minimal</u> A_6 équivalent.

Solution : On complète l'automate et on sépare les états terminaux et le seul état non terminal, car Θ_0 ={T, NT} où T={0,1,2,6,4,5}, NT={P}

	état	a	b	sous Θ_0	
T	0	P	P	NT	NT
	1	3	5	T	T
	2	4	0	T	T
	3	0	1	T	T
	4	3	5	T	T
	5	4	0	T	T
NT	P	P	P		

L'itération 1 (sous Θ_0) donne Θ_1 ={(0), A, (P)} où A={1,2,6,4,5}

L'itération 2 (sous Θ_1) :

état	a	b	sous Θ_1	
1	3	5	A	A
2	4	0	A	0
3	0	1	0	A
4	3	5	A	A
5	4	0	A	0

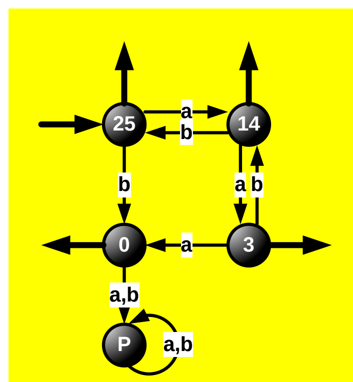
On obtient Θ_2 ={(0), (1,4), (2,5), (3), (P)} . L'itération 3 :

			sous Θ_2		
1	3	5	3	2 5	pas de séparation
4	3	5	3	2 5	
2	4	0	1 4	0	pas de séparation
5	4	0	1 4	0	

Donc $\Theta_3 = \Theta_2 = \Theta_{fin}$

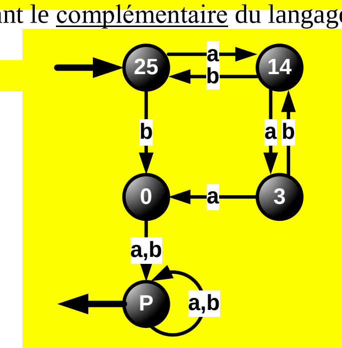
L'entrée : groupe (2,5). Les sorties : tous les groupes sauf P. Les transitions :

S	0	P	P
S	1 4	3	2 5
E/S	2 5	1 4	0
S	3	0	1 4
	P	P	P



b) Construire un automate <u>déterministe complet minimal</u> A_7 reconnaissant le <u>complémentaire</u> du langage reconnu par l'automate A_5.

Solution : il suffit de complémentariser l'automate minimal obtenu



Exercice 5

a) Construire un automate reconnaissant le langage

$$L = \{((a+b)(a+b))^* + ((a+b)(a+b)(a+b)(a+b))^* \}$$

suivant les règles formelles données dans le cours.

Puis, au choix,

soit

b) déterminiser et
c) minimiser

l'automate obtenu en (a) (c'est assez compliqué !),

soit (c'est plus facile à réaliser)

b) simplifier graphiquement et déterminiser, puis
c) minimiser

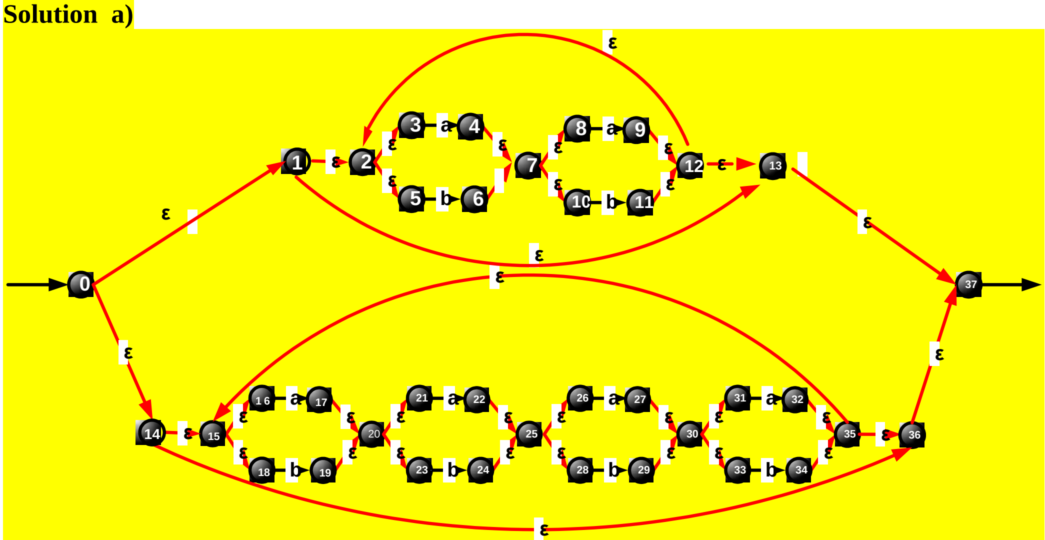
l'automate simplifié graphiquement.

soit

(b+c) produire directement l'automate déterministe complet minimal que vous devriez obtenir en (c), si vous savez le faire *<u>et si vous savez l'expliquer</u>*. Dans ce cas, vous n'êtes pas tenus de faire le (b), mais vous devez toujours faire le (a).

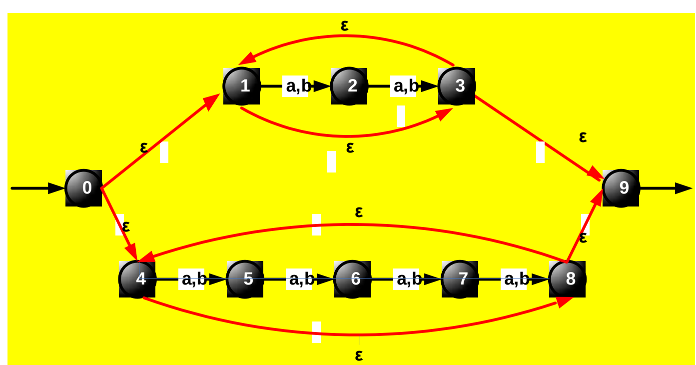
Notation pour cette exo : a) 33.3%, b) 33.3%, c) 33.3%.

Solution a)

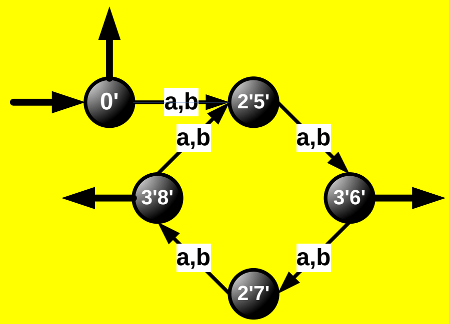


b) Je choisis de déterminiser un automate graphiquement simplifié :

	notation complète	a ou b		notation en ε-clôtures	a ou b
E/S	013489	25	E/S	0'	2'5'
	25	1369		2'5'	3'6'
S	1369	27	S	3'6'	2'7'
	27	13489		2'7'	3'8'
S	13489	25	S	3'8'	2'5'



L'automate déterministe obtenu :

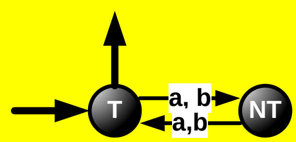


d) Lors de la minimisation, il ne se produit aucune séparation :

T	0'	2'5'	NT
	3'6'	2'7'	NT
	3'8'	2'5'	NT

NT	2'5'	3'6'	T
	2'7'	3'8'	T

Donc l'automate minimal n'a que deux états :



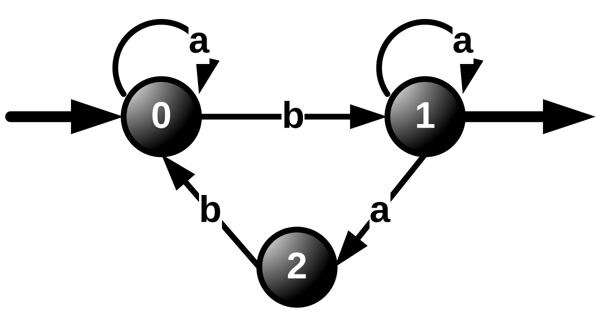
On aurait pu éviter le processus de déterminisation et minimisation, en arguant que le langage

$$L = \{((a+b)(a+b))^* + ((a+b)(a+b)(a+b)(a+b))^*\} = \{((a+b)(a+b))^*\}$$

car tous les multiples de 4 font partie des multiples de 2, et donc l'automate minimal est un cycle de longueur 2 avec la sortie en position 0.

Exercice 6

Trouver le langage reconnu par l'automate suivant :



Vous pouvez utiliser soit la méthode de l'arrivée, soit celle d'élimination d'états. Aucune solution basée sur de « l'intuition » ne sera acceptée.

Solution : Par la méthode de l'arrivée :

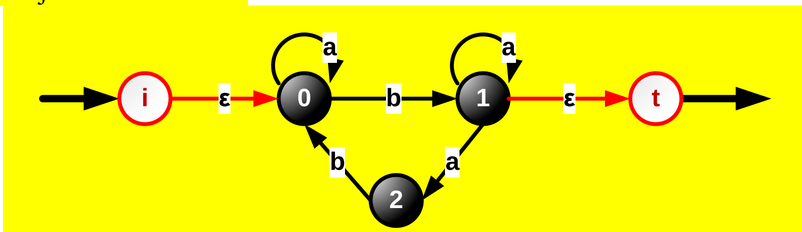
$$\begin{cases} 0 = \varepsilon + 0a + 2b & (1) \\ 1 = 0b + 1a & (2) \\ 2 = 1a & (3) \end{cases}$$

L=1

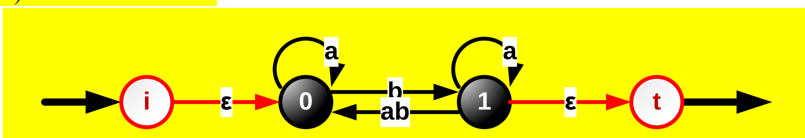
(2) $\Rightarrow$ 1=0ba*, et donc 2=1a=0ba*a=0ba$^+$. Remplaçant dans (1), on obtient 0 = ε + 0a + 0ba$^+$b = ε + 0(a+ ba$^+$b), donc 0 = ε(a+ ba$^+$b)*= (a+ ba$^+$b)*. Donc 1=0ba*=(a+ ba$^+$b)*ba* = L.

Par la méthode d'élimination d'états :

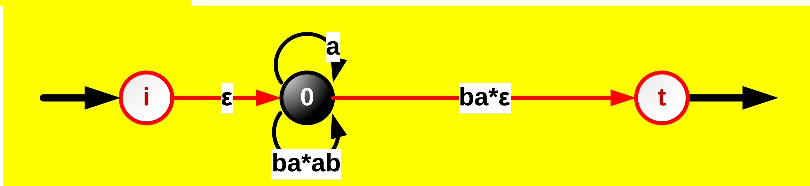
1) L'ajout des états i et t :



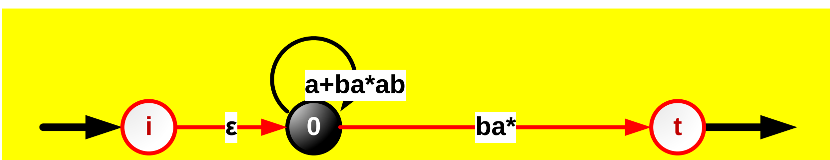
2) a) élimination de 2 :



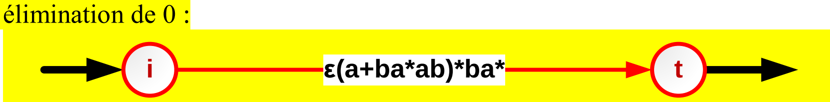
b) élimination de 1 :



ce qui est la même chose que



c) élimination de 0 :



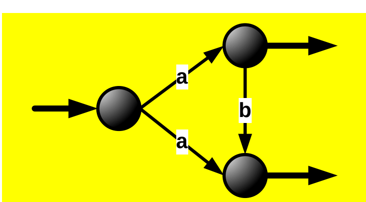
Donc L = (a+ba*ab)*ba*.

On peut obtenir, en changeant l'ordre des opérations, d'autres expressions rationnelles dont l'égalité à celle-ci n'est pas immédiatement evidente.

Questions de cours

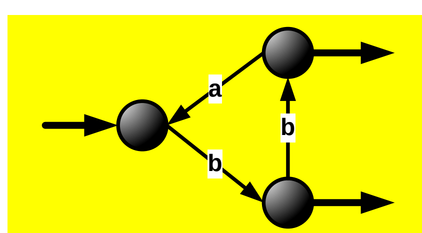
1. Un automate standard est-il obligatoirement déterministe ? Si la réponse est non, donner un exemple d'automate standard non déterministe.

   Non, voici un exemple :

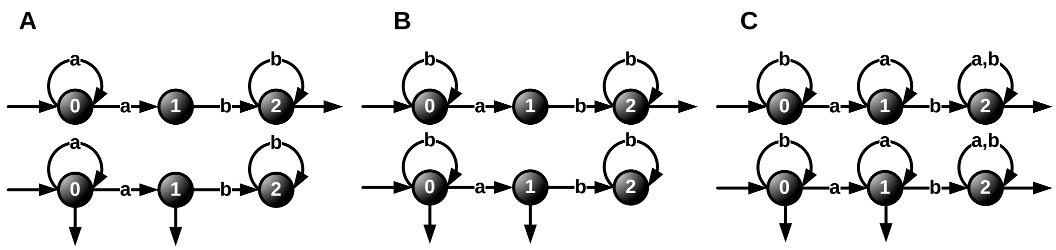


2. Un automate déterministe est-il obligatoirement standard? Si la réponse est non, donner un exemple d'automate déterministe non standard.

   Non, voici un exemple :



3. Un automate complet contient-il toujours un état poubelle (entourer la bonne réponse) **NON**
4. Combien il y a-t-il d'états initiaux dans un automate minimal? Un seul
5. Combien il y a- t-il d'états terminaux dans un automate minimal? Jusqu'au nombre d'états moins un, à l'exception de l'automate minimal reconnaissant A*, où le seul état de l'automate est une entrée/sortie. **Attention :** j'ai accepté ici des formulations en réalité absurdes, comme « un nombre infini » (?!), « autant qu'on veut » (qui ? Et si j'en veux mille ?), « on ne peut pas savoir » (minimisez, et vous le saurez), « ce nombre n'est pas défini » (idem) etc.
6. Pour chacun des couples d'automates A, B et C, dire si l'automate en haut reconnaît le langage complémentaire à celui reconnu par l'automate en bas. Expliquer le oui et le non en phrases courtes. La réponse vous donne des points uniquement si toutes les trois réponses sont bonnes.



A: non, car l'automate du haut n'est pas déterministe. Les deux ne reconnaissent pas le mot 'b'.
B : non, car l'automate du haut n'est pas complet. Les deux ne reconnaissent pas le mot 'aa'.
C : non, car si l'automate du haut est un ADC, la sortie sur 2 figure toujours dans l'automate du bas. Les deux automates reconnaissent 'ab'.