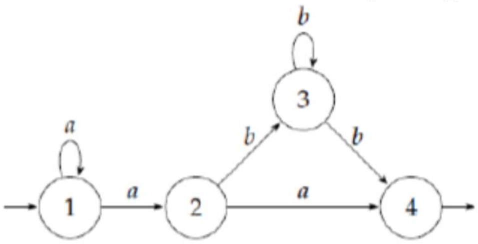
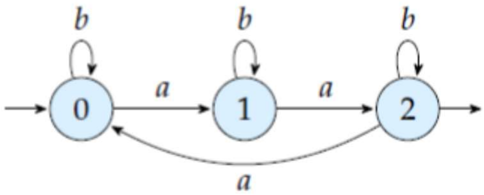
TD_ opérations sur les automates

Automate fini déterministe AFD

Définition : Un AEF (A,Q, q0, QF, δ) est dit **déterministe** si les deux conditions sont vérifiées :

- $\forall qi \in Q, \forall a \in X$, il existe au plus un état qj tel que $\delta(qi, a) = qj$;
- L'automate ne comporte pas de ε-transitions.





Algorithme : Déterminiser un AEF sans les ε-transitions

dans l'algorithme suivant, chaque ensemble d'états représente un état du futur automate.

1- Partir de l'état initial $E^{(0)} = \{q_0\}$ (c'est l'état initial du nouvel automate) ;

2- Construire $E^{(1)}$ l'ensemble des états obtenus à partir de $E^{(0)}$ par la transition a :

$$E^{(1)} = \bigcup_{q' \in E(0)} \delta(q', a)$$

3- Recommencer l'étape 2 pour toutes les transitions possibles et pour chaque nouvel ensemble $E^{(i)}$;

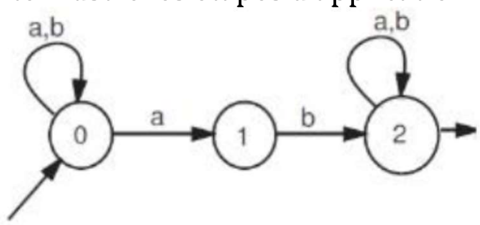
$$E^{(i)} = \bigcup_{q' \in E(i-1)} \delta(q', a)$$

4- Tous les ensembles contenant au moins un état final du premier automate deviennent finaux ;

5- Renuméroter les états en tant qu'états simples.

Pour illustrer cet algorithme, nous allons l'appliquer à l'automate suivant

La table suivante illustre les étapes d'application de l'algorithme



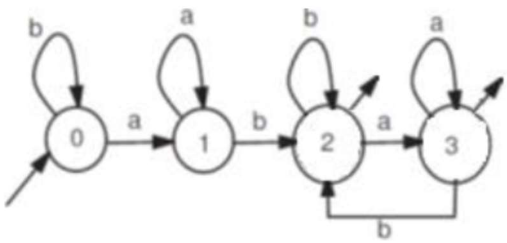
État	a	b
0	0,1	0
0,1	0,1	0,2
0,2	0,1,2	0,2
0,1,2	0,1,2	0.2

⟹

État	a	b
0	1	0
1	1	2
2	3	2
3	3	2

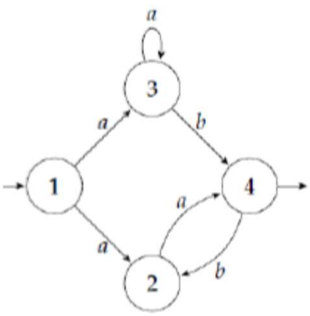
L'état initial : 0
les états finaux: {0,2},{0,1,2}
(parce que 2 est un état finale dans l'automate originale)

L'automate déterministe est le suivant

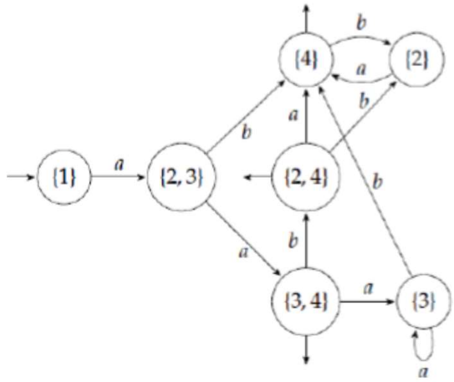


Exemple 2 :

Un automate à déterminiser



Le résultat de la déterminisation



Minimisation d'un AEF déterministe

La méthode de réduction d'un AEF est la suivante :

1. *Nettoyer* l'automate en éliminant les états inaccessibles ;
   Un état est dit *inaccessible* s'il n'existe aucun chemin permettant de l'atteindre à partir de l'état initial. (c'est-à-dire qu'ils ne participe jamais à la reconnaissance d'un mot).
2. Regrouper les états congruents (appartenant à la même classe d'équivalence).

Algorithme : Regroupement des états congruents

1- Faire deux classes : A contenant les états finaux et B contenant les états non finaux ;
2- S'il existe un symbole a et deux états q_i et q_j d'une même classe tel que $\delta(q_i, a)$ et $\delta(q_j, a)$ n'appartiennent pas à la même classe, alors créer une nouvelle classe et séparer q_i et q_j. On laisse dans la même classe tous les états qui donnent un état d'arrivée dans la même classe ;
3- Recommencer l'étape 2 jusqu'à ce qu'il n'y ait plus de classes à séparer ;

Exemple : Soit à minimiser l'automate suivant (les états finaux sont les états 1 et 2 tandis que l'état 1 est initial) :

État	a	b
1	2	5
2	2	4
3	3	2
4	5	3
5	4	6
6	6	1
7	5	7

1. La première étape consiste à *éliminer les états inaccessibles*, il s'agit juste de l'état 7.
   Les étapes de détermination des classes de congruences sont les suivantes :
   A = {1, 2}, B = {3, 4, 5, 6} ;

2. $\delta(3, b) = 2 \in A$,$\delta(4, b) = 3 \in B$ ainsi il faut séparer 4 du reste de la classe B. Alors, oncrée une classe C contenant l'état 4 ; (A = {1, 2}, C = {4}, B = {3, 5, 6})
   $\delta(3, b) = 2 \in A$, $\delta(5, b) = 6 \in A$ ainsi il faut séparer 5 du reste de la classe B.
   Mais inutile de créer une autre classe puisque
   $\delta(4, a) = 5 \in B$,
   $\delta(5, a) = 4 \in B$
   et $\delta(4, b) = 3 \in B$
   et $\delta(5, b) = 6 \in B$,

   il faut donc mettre 5 dans la classe C. C = {4, 5} et B = {3, 6} ;
   (A = {1, 2}, C = {4, 5} et B = {3, 6})

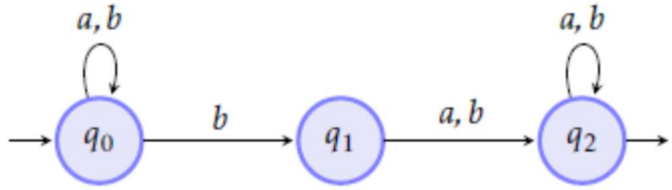
4. Aucun autre changement n'est possible, alors on arrête l'algorithme.

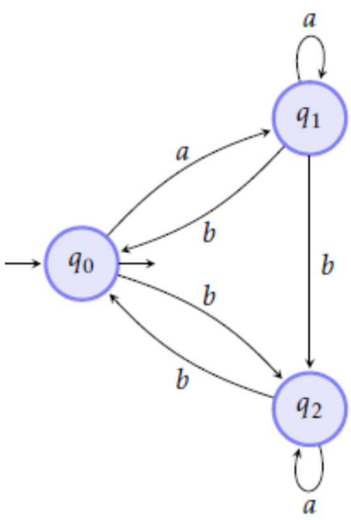
Le nouvel automate est donc le suivant (l'état initial est A) :

Etat	**a**	**b**
A	A	C
B	B	A
C	C	A

L'automate obtenu est **minimal** et est **unique**,

Exercice 01: déterminisation
Déterminiser les automates suivants :





Exercice 02 : opérations sur les automates

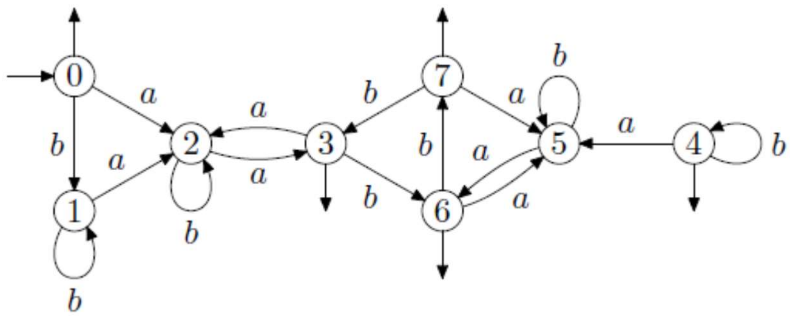
Construisez des automates finis **déterministes** acceptant le langage décrit dans chacun des cas suivants :

1. L_1 = {w ∈ {a, b}* | w contient ***bab*** }.
2. L_2 = {w ∈ {a, b}* | w contient ***bb*** }.
3. L_3 = {w ∈ {a, b}* | w ne contient pas ***bab*** }
4. L_4 = {w ∈ {a, b}* | w ne contient pas ***bb*** }
5. L_5 = {w ∈ {a, b}* | w contient ***bab*** ou ***ba*** }.
6. L_6 = {w ∈ {a, b}* | w ne contient ni ***bab*** ni ***bb***}.
7. L_7 = {w ∈ {a, b}* | w contient à la fois ***bab*** et ***bb*** }.

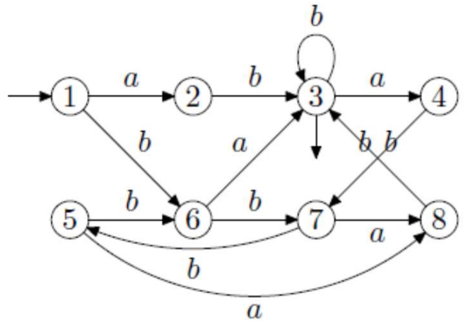
Exercice 02 : minimisation des automates

Minimiser les automates suivants

L'automate A



L'automate B

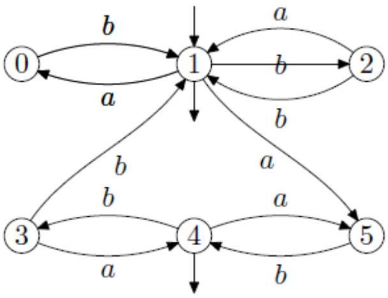


Exercice 04 : Comparer entre eux des expressions rationnelles et des automates

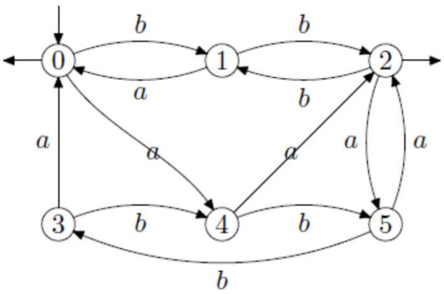
Parmi les expressions rationnelles et les automates suivants dire quels sont les automates et les expressions rationnelles qui représentent le même langage.

$(ab^*a + b(a+b))^*$

$(ab + b(a+b))^*$



L'automate A1



L'automate A1