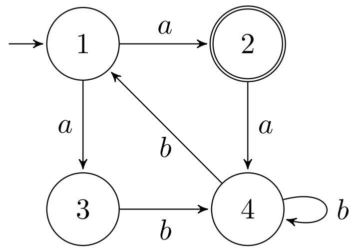
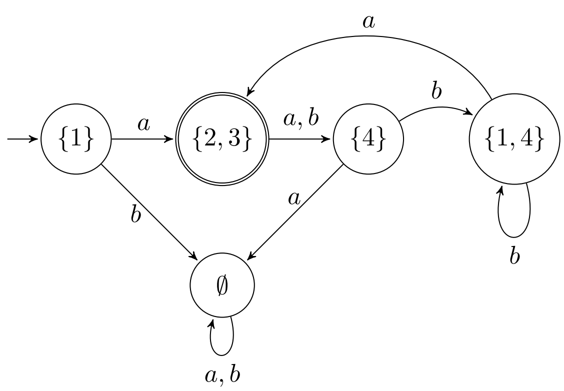
I Algorithme de déterminisation

Déterminiser l'automate suivant en utilisant l'algorithme du cours :



Solution : En appliquant l'algorithme du cours, on obtient :



II Clôture des langages reconnaissables

Si $m = m_1 ... m_n$ est un mot, on définit son miroir $\widetilde{m} = m_n ... m_1$. Si L est un langage, on définit son miroir $\widetilde{L} = \{\widetilde{m} \mid m \in L\}$.

1. Montrer que le miroir d'un langage reconnaissable est reconnaissable.

   Solution : Soit $A = (\Sigma, Q, I, F, E)$ un langage reconnaissant L. Alors $\widetilde{A} = (\Sigma, Q, F, I, \widetilde{E})$ reconnaît $\widetilde{L}$, où on a inversé toutes les transitions ($\widetilde{E} = \{(q_1, a, q_2) \mid (q_2, a, q_1) \in E\}$). En effet il existe un chemin $q_1 \in I \xrightarrow{m_1} q_2 \xrightarrow{m_2} ... \xrightarrow{m_k} q_{k+1} \in F$ dans A si et seulement s'il existe un chemin $q_{k+1} \in F \xrightarrow{m_k} q_k \xrightarrow{m_{k-1}} ... \xrightarrow{m_1} q_1 \in I$ dans $\widetilde{A}$.

Si L est un langage sur Σ, on définit :

- $Pref(L) = \{u \in \Sigma^* \mid \exists v \in \Sigma^*,\ uv \in L\}$: ensemble des préfixes des mots de L.
- $Suff(L) = \{u \in \Sigma^* \mid \exists v \in \Sigma^*,\ vu \in L\}$: ensemble des suffixes des mots de L.
- $Fact(L) = \{u \in \Sigma^* \mid \exists v, w \in \Sigma^*,\ vuw \in L\}$: ensemble des facteurs des mots de L.

2. Donner des expressions régulières pour $Pref(a^*b)$ et $Pref((ab)^*)$.

   Solution : $Pref(a^*b) = a^*|a^*b$, $Pref((ab)^*) = (ab)^*|(ba)^*|(ab)^*a|(ba)^*b$.

3. Montrer que si L est reconnaissable alors $Pref(L)$, $Suff(L)$, $Fact(L)$ le sont aussi.

   Solution : Soit $A = (\Sigma, Q, I, F, E)$ un langage reconnaissant L. Soit Q' l'ensemble des états co-accessibles et Q'' l'ensemble des états accessibles. Un mot m appartient à $Pref(L)$ si et seulement si il existe un chemin étiqueté par m d'un état de I vers un état de Q'. Donc (Σ, Q, I, Q', E) reconnaît $Pref(L)$. De même, (Σ, Q, Q'', F, E) reconnaît $Suff(L)$ et (Σ, Q, Q'', Q', E) reconnaît $Fact(L)$.

   Autre solution : après avoir démontré que $Pref(L)$ est reconnaissable, on peut remarquer que $Suff(L) = \widetilde{Pref(\widetilde{L})}$
   ($m \in \widetilde{Pref(\widetilde{L})} \iff \widetilde{m} \in Pref(\widetilde{L}) \iff \exists v \in \Sigma^*,\ \widetilde{m}v \in \widetilde{L} \iff \exists v \in \Sigma^*,\ \widetilde{v}m \in L \iff m \in Suff(L)$, où on a utilisé le fait que $\widetilde{xy} = \widetilde{y}\ \widetilde{x}$).
   On peut aussi en déduire que $Fact(L)$ est reconnaissable en remarquant que $Fact(L) = Suff(Pref(L))$ ($= Pref(Suff(L))$).
   En effet $m \in Suff(Pref(L)) \iff \exists u \in \Sigma^*,\ um \in Pref(L) \iff \exists u \in \Sigma^*,\ \exists v \in \Sigma^*,\ umv \in L \iff m \in Fact(L)$.

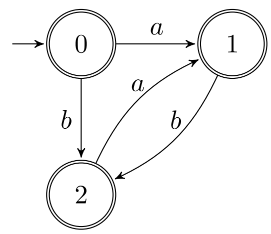
4. Montrer que si L est régulier alors $Pref(L)$, $Suff(L)$, $Fact(L)$ le sont aussi (puisqu'on va montrer que régulier = reconnaissable, c'est une preuve alternative à la précédente).

> Solution : Soit e est une expression régulière dont le langage est L. On définit par induction une expression régulière $P(e)$ de langage $Pref(L)$:
>
> - Si $e = a \in \Sigma$: $P(e) = \varepsilon + a$.
> - Si $e = e_1|e_2$: $P(e) = P(e_1)|P(e_2)$.
> - Si $e = e_1e_2$: $P(e) = P(e_1)|e_1P(e_2)$.
> - Si $e = e_1^*$: $P(e) = e_1^*P(e_1)$.
>
> De même pour $Suff(L)$ et $Fact(L)$.

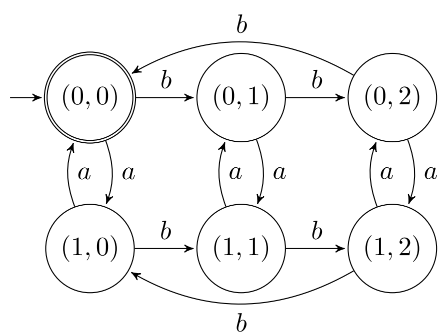
III Reconnaissable ou non ?

Pour chacun de ces langages, dire s'il est reconnaissable ou non. Justifier.

1. L_1 = mots sur $\{a, b\}$ sans lettres consécutives égales.

> Solution : on se retrouve dans l'état 1 (respectivement 2) si la dernière lettre lue est un a (resp. b).
>
> 
>

2. L_2 = mots sur $\{a, b\}$ ayant un nombre pair de a et dont le nombre de b est multiple de 3.

> Solution : On peut construire un automate reconnaissant les mots ayant un nombre pair de a en utilisant 2 états (suivant la parité du nombre de a lus jusqu'à présent). De façon similaire, les mots ayant un nombre de b multiple de 3 peuvent être reconnus par un automate avec 3 états, pour chaque reste modulo 3 du nombre de b déjà lus.
> Pour vérifier les deux en même temps, on peut utiliser l'automate « produit » vu en cours, pour reconnaître l'intersection des deux langages précédents. L'état nommé (i, j) correspond à la lecture d'un mot dont le nombre de a est i modulo 2 et le nombre de b est j modulo 3 :
>
> 
>

3. $L_3 = \{u \in \{a, b\}^* \mid |u|_a \mod 2 = |u|_b \mod 3\}$ (où $|m|_a$ est le nombre de a du mot m).

> Solution : Reprendre l'automate précédent en choisissant $(0, 0)$ et $(1, 1)$ comme états finaux.

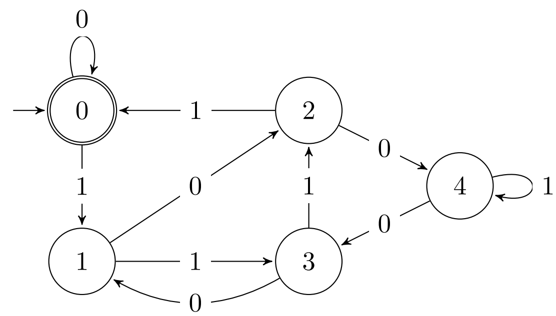
4. $L_4 = \{m \in \{a, b\}^* \mid |m|_a = |m|_b\}$.

> Solution : Supposons que L_4 soit reconnaissable. Soit n l'entier donné par le lemme de l'étoile et $u = a^n b^n$.
> $u \in L_4$ et $|u| \geq n$ donc, par le lemme de l'étoile, il existe des mots x, y, z tels que $|xy| \leq n$, $y \neq \varepsilon$ et $\forall k \in \mathbb{N}$, $xy^k z \in L$.
> Comme $|xy| \leq n$, y ne contient que des a. En prenant $k = 0$, on trouve que $xy^0 z = xz \in L_3$. Mais xz contient

strictement plus de b que de a, ce qui est absurde.

5. L_5 = écritures en base 2 des multiples de 5.

Solution : On va construire un automate dont les états sont les restes possibles modulo 5. Si l'automate est dans un état q a déjà lu $n_1...n_p$ (c'est à dire $\overline{n_1...n_p}^2 \equiv q[5]$) et qu'il lit n_{p+1} alors il doit aller dans l'état $2q + n_{p+1}$ car $\overline{n_1...n_p n_{p+1}}^2 = 2 \times \overline{n_1...n_p}^2 + n_{p+1} \equiv 2q + n_{p+1}[5]$.
L_5 est donc reconnu par l'automate $(\Sigma, Q, I, F, \delta) = (\{0,1\}, \{0,1,2,3,4\}, \{0\}, \{0\}, \delta)$ où $\delta(q,a) = 2q + a\ [5]$:



6. $L_6 = \{a^p \mid p \text{ est un nombre premier}\}$.

Solution : Supposons que L_6 soit reconnaissable. Soit n l'entier donné par le lemme de l'étoile.
Soit $p \geq n+2$ un nombre premier et $u = a^p$.
$u \in L$ et $|u| \geq n$ donc, par le lemme de l'étoile, il existe i_1, i_2, i_3 tels que $i_1 + i_2 \leq n$, $i_2 > 0$ et $\forall k \in \mathbb{N}$, $a^{i_1}(a^{i_2})^k a^{i_3} = a^{i_1+ki_2+i_3} \in L$ (on a utilisé le fait que u ne contient que des a).
Soit $k = i_1 + i_3$ et $q = i_1 + ki_2 + i_3$. Alors, d'après le lemme de l'étoile, $a^q \in L$. Mais $q = (i_1+i_3)(1+i_2)$ avec $1 + i_2 > 1$ et $i_1 + i_3 \geq i_3 > 1$ (car $p \geq n+2$) donc q n'est pas premier, ce qui est absurde.

IV Algorithmes sur les automates

1. À quelle condition nécessaire et suffisante simple le langage reconnu par un automate est vide ? Décrire un algorithme pour le savoir.

Solution : Il est vide si et seulement si il n'existe pas de chemin d'un état initial vers un état final. On peut le décider en effectuant un parcours du graphe de l'automate.

2. À quelle condition nécessaire et suffisante simple le langage reconnu par un automate est fini ? Décrire un algorithme pour le savoir.

Solution : Il est fini si et seulement si son graphe n'a pas de cycle. Nous avons vu un algorithme pour cela dans le cours sur les graphes.

3. Décrire un algorithme pour déterminer si deux automates admettent le même langage.

Solution : Soient $A_1 = (\Sigma, Q_1, i_1, F_1, \delta_1)$ et $A_2 = (\Sigma, Q_2, i_2, F_2, \delta_2)$. On peut construire des automates A et A' reconnaissant $L(A_1)\backslash L(A_2)$ et $L(A_2)\backslash L(A_1)$, en utilisant « l'automate produit » décrit dans le cours. Il suffit alors de déterminer si $L(A) = \emptyset$ et $L(A') = \emptyset$, en utilisant la question 3.

V Longueur discriminante

1. Soit A un automate. Décrire un algorithme pour déterminer la plus petite longueur d'un mot reconnu par A et préciser sa complexité.
2. Soit A un automate à n états et de langage $L(A)$. Montrer que $L(A) = \emptyset$ si et seulement si $L(A)$ ne contient aucun mot de longueur strictement inférieure à n.
3. Soit $A_1 = (Q_1, i_1, F_1, \delta_1)$ et $A_2 = (Q_2, i_2, F_2, \delta_2)$ deux automates déterministes complets à n_1 et n_2 états et de langages L_1 et L_2. On suppose que $L_1 \neq L_2$. Soit $l(L_1, L_2)$ la plus petite longueur d'un mot u appartenant à l'un des deux langages mais pas à l'autre.
Montrer que $l(L_1, L_2) < n_1 n_2$.

Solution :

1. On peut effectuer un parcours en largeur depuis les états initiaux de A pour trouver la longueur d'un plus court chemin vers un état final. Si la fonction de transition de l'automate est donnée par une liste d'adjacence, la complexité est en $\mathrm{O}(n + p)$ où n est le nombre d'états de A et p le nombre de transitions.
2. $\Rightarrow$: Évident.
$\Leftarrow$: Supposons que $L(A)$ ne contienne aucun mot de longueur inférieure ou égale à $n - 1$.
Supposons par l'absurde que $L(A) \neq \emptyset$ et soit $u \in L(A)$ de taille minimum.
Comme $|u| \geqslant n$, u est l'étiquette d'un chemin acceptant C passant par au moins $n + 1$ états (car il y a $n + 1$ sommets dans un chemin à n sommets).
D'après le principe des tiroirs, C passe deux fois par le même sommet.
En supprimant ce cycle, on obtient un chemin acceptant pour un mot de taille strictement inférieure à $|u|$, ce qui est absurde.
Donc $L(A) = \emptyset$.
3. Comme $L_1 \neq L_2$, $L_1 \Delta \overline{L_2} \neq \emptyset$.
Soit A l'automate $\big(Q_1 \times Q_2, (i_1, i_2), F, \delta\big)$, où $\delta((q_1, q_2), a) = (\delta_1(q_1, a), \delta_2(q_2, a))$ et $F = \big(F_1 \times \overline{F_2}\big) \cup \big(\overline{F_1} \times F_2\big)$.
Alors A reconnaît $L_1 \Delta L_2$ et possède $n_1 n_2$ états donc, d'après la question 22), $L_1 \Delta L_2$ contient un mot de longueur strictement inférieure à $n_1 n_2$.

VI Ensemble distingant

Soient L un langage sur un alphabet Σ et $u, v \in \Sigma^*$. On dit que $w \in \Sigma^*$ est un *suffixe distingant* pour u et v si exactement l'un des mots uw ou vw appartient à L.

Un ensemble de mots D est *distingant* pour L si toute paire de mots de D a un suffixe distingant.

1. Soit L_1 le langage dénoté par l'expression régulière $(ab)^*$. Montrer que $\{\varepsilon, a, b\}$ est un ensemble distingant pour L_1.
2. On note $ind(L)$ le nombre minimum d'états d'un automate déterministe complet reconnaissant L. Montrer que si L a un ensemble distingant de taille n alors $ind(L) \geqslant n$.
3. Que vaut $ind(L_1)$?
4. On suppose que L a un ensemble distingant infini. Montrer que L n'est pas un langage régulier.
5. En déduire que $\{a^n b^n \mid n \in \mathbb{N}\}$ n'est pas un langage régulier.
6. Soit L_2 l'ensemble des mots de $\{a, b\}^*$ qui contiennent un nombre pair de a et un nombre pair de b. Déterminer $ind(L_2)$.

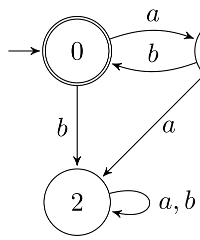
Solution :

1.
   - b est un suffixe distingant pour ε et a car $\varepsilon b \notin L_1$ et $ab \in L_1$.
   - ab est un suffixe distingant pour ε et b car $\varepsilon ab \in L_1$ et $bab \in L_1$.
   - b est un suffixe distingant pour a et b car $ab \in L_1$ et $bb \notin L_1$.

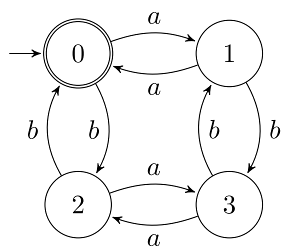
   Donc $\{\varepsilon, a, b\}$ est un ensemble distingant pour L_1.
2. Supposons que L ait un ensemble distingant D de taille n et qu'il existe un automate déterministe complet A à $k < n$ états reconnaissant L. Soit q_0 l'état initial de A. Par le principe des tiroirs, il existe $u, v \in D$ tels que $\delta^*(q_0, u) = \delta^*(q_0, v)$. Soit w un suffixe distingant pour u et v. Alors $\delta^*(q_0, uw)$ est un état final mais pas $\delta^*(q_0, vw)$,

ce qui est absurde.

3. Comme $\{\varepsilon, a, b\}$ est distingant et que l'automate déterministe complet ci-dessous reconnaît L_1, on a $ind(L_1) = 3$.



4. Si L est régulier alors $ind(L) \geqslant n$ pour tout $n \in \mathbb{N}$: absurde.
5. a^* est un ensemble distingant infini (si $i \neq j$ alors b^i est un suffixe distingant pour a^i et a^j) donc n'est pas régulier d'après la question 4.
6. On montre que L_2 a un ensemble distingant de taille 4 : $\{\varepsilon, a, b, ab\}$.
De plus, l'automate déterministe complet suivant à 4 états reconnaît L_2 :



Donc $ind(L_2) = 4$.

VII Résiduel

Soit L un langage sur un alphabet Σ. Soit $u \in \Sigma^*$. On définit le langage $u^{-1}L = \{v \in \Sigma^* \mid uv \in L\}$ (qu'on appelle résiduel de L).

1. Montrer que si L est reconnaissable alors $u^{-1}L$ est reconnaissable.

> Solution : Soit $A = (\Sigma, Q, i, F, \delta)$ un automate déterministe complet reconnaissant L (où i est l'état initial) et q_u l'état obtenu en lisant u dans A (c'est à dire $q_u = \delta^*(i, u)$). Alors : $m \in u^{-1}L \iff um \in L \iff$ dans A, il existe un chemin étiqueté par um de i vers un état final $\iff$ dans A, il existe un chemin étiqueté par m de q_u vers un état final. Donc $(\Sigma, Q, q_u, F, \delta)$ reconnaît $u^{-1}L$.
> Remarque : il n'est pas obligatoire de considérer un automate déterministe complet, mais cela simplifie légèrement la preuve.

2. Quels sont tous les résiduels possibles de a^*b^*? De $\{a^n \mid n \text{ est pair}\}$?

> Solution : Soit u un mot sur $\{a, b\}$. Si u contient un facteur ba alors $u^{-1}L(a^*b^*) = \emptyset$. Si u ne contient que des a alors $u^{-1}L(a^*b^*) = L(a^*b^*)$. Si u ne contient un certain nombre de a suivit d'un certain nombre de b (mais au moins un b), alors $u^{-1}L(a^*b^*) = L(b^*)$. Les résiduels possibles de $L(a^*b^*)$ sont donc $\emptyset$, $L(b^*)$, $L(a^*b^*)$.
> Les résiduels possibles de $\{a^n \mid n \text{ est pair}\}$ sont $\{a^n \mid n \text{ est pair}\}$ et $\{a^n \mid n \text{ est impair}\}$ (suivant que u contienne un nombre pair ou impair de a).

3. Montrer que si L est reconnaissable alors $\{u^{-1}L \mid u \in \Sigma^*\}$ est fini (il n'y a qu'un nombre fini de valeurs possibles pour $u^{-1}L$ quand u varie dans Σ^*).

> Solution : Soit $A = (\Sigma, Q, i, F, \delta)$ un automate déterministe complet reconnaissant L (le même que pour la question précédente). Soit $n = |Q|$. Supposons qu'il existe $n+1$ résiduels différents, notés $u_0^{-1}L$, $u_1^{-1}L$, ..., $u_n^{-1}L$. Comme il y a n états dans A, les états $\delta^*(i, u_0), ..., \delta^*(i, u_n)$ ne peuvent pas tous être différents: disons par exemple $\delta^*(i, u_k) = \delta^*(i, u_\ell)$. Mais alors $u_k^{-1}L$ et $u_\ell^{-1}L$ sont reconnus par le même automate donc sont égaux, ce qui est absurde. Donc il y a au plus $|Q|$ langages résiduels différents.

4. Montrer que $\{a^n b^n \mid n \in \mathbb{N}\}$ n'est pas reconnaissable.

> Solution : Soit $k \in \mathbb{N}$. Alors $b^k \in (a^k)^{-1}L$ mais $\forall p \neq k$, $b^k \notin (a^p)^{-1}L$. Donc L possède une infinité de résiduels différents: $(a^k)^{-1}L$ pour $k \in \mathbb{N}$. Par contraposition de la question précédente, L n'est donc pas reconnaissable.

Un mot m est un palindrome s'il se lit de la même façon dans les deux sens (ou encore: $\widetilde{m} = m$).

5. Montrer que l'ensemble des palindromes (sur un alphabet à au moins 2 lettres) n'est pas reconnaissable.

> Solution : Soit L le langage des palindromes sur un alphabet Σ et a, b deux lettres de Σ. Alors $a^k bb \in (a^k)^{-1}L$ mais $\forall p \neq k$, $a^k bb \notin (a^p)^{-1}L$. Donc L a une infinité de résiduels différents donc n'est pas reconnaissable.