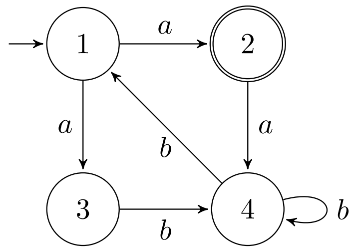
I Algorithme de déterminisation

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



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.

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

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.
2. L_2 = mots sur $\{a, b\}$ ayant un nombre pair de a et dont le nombre de b est multiple de 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).
4. $L_4 = \{m \in \{a, b\}^* \mid |m|_a = |m|_b\}$.
5. L_5 = écritures en base 2 des multiples de 5.
6. $L_6 = \{a^p \mid p \text{ est un nombre premier}\}$.

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.
2. À quelle condition nécessaire et suffisante simple le langage reconnu par un automate est fini ? Décrire un algorithme pour le savoir.
3. Décrire un algorithme pour déterminer si deux automates admettent le même langage.

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 $A_1 = (\Sigma, Q_1, i_1, F_1, \delta_1)$ et $A_2 = (\Sigma, Q_2, i_2, F_2, \delta_2)$ 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 $l(L_1, L_2) < n_1 n_2$.

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 $ind(L_2)$.

VII Automate des palindromes (oral ENS info)

On fixe un alphabet Σ avec $|\Sigma| > 1$. Un mot $w \in \Sigma^*$ est un palindrome s'il s'écrit $w = a_1 \cdots a_n$ et qu'on a $a_i = a_{n-i+1}$ pour tout $1 \leq i \leq n$. On note $\Pi \subseteq \Sigma^*$ le langage des palindromes. Pour un automate fini A sur Σ, on note $L(A)$ le langage reconnu par A.

1. Soit $\Pi_n := \Pi \cap \Sigma^n$. Montrer que pour tout automate fini déterministe complet A, pour tout $n \in \mathbb{N}$, si $L(A) \cap \Sigma^{2n} = \Pi_{2n}$, alors A a au moins $|\Sigma|^n$ états.
2. En déduire que le langage Π n'est pas reconnaissable.
3. Étant donné un automate fini A sur Σ, peut-on calculer un automate A_Π qui reconnaisse $L(A) \cap \Pi$?
4. Pour tout mot $u = b_1 \cdots b_m$ de Σ^*, on note $\overline{u} := b_m \cdots b_1$ son miroir. Étant donné A, peut-on calculer un automate A'_Π qui reconnaisse $\{u \in \Sigma^* \mid u\overline{u} \in L(A)\}$?
5. On appelle Π_{pair} l'ensemble des palindromes de longueur paire, i.e., $\Pi_{\text{pair}} = \bigcup_{n \in \mathbb{N}} \Pi_{2n}$. Proposer un algorithme qui, étant donné un automate fini A sur Σ, détermine si $L(A) \cap \Pi_{\text{pair}}$ est vide, fini, ou infini. Discuter de sa complexité en temps et en espace.
6. Modifier l'algorithme de la question 4 pour calculer la cardinalité de $L(A) \cap \Pi_{\text{pair}}$ quand cet ensemble est fini, en faisant l'hypothèse que l'automate d'entrée A est déterministe. Comment la complexité est-elle affectée?
7. Modifier l'algorithme des questions 4 et 5 pour qu'il s'applique à $L(A) \cap \Pi$.