Le nombre de symboles indique la difficulté de l'exercice. Le devoir est noté sur plus que 20, il est donc techniquement possible d'avoir 20 en n'ayant pas tout résolu. Sélectionnez donc les exercices que vous voulez résoudre en priorité.

Exercice 1 : langages vers expressions régulières et automates

Considérons l'alphabet $\Sigma = \{a, b\}$. Pour chacun des langages suivants sur Σ, donner une expression régulière qui le dénote et un automate qui le reconnait.

1. L'ensemble des mots qui contiennent le facteur ab et finissent par a.
2. L'ensemble des mots qui contiennent exactement deux b.
3. L'ensemble des mots qui ne sont pas $\{a, b, ab\}$.
4. L'ensemble des mots w tels que $|w|_a - |w|_b \equiv 1[3]$ (*i.e.* le nombre de a moins le nombre de b est congru à 1 modulo 3) et $|w|_a \equiv 0[2]$.

Exercice 2 : Régularité d'un langage

Prouver que les langages suivants ne sont pas réguliers.

1. $\{w \mid w \in \{a, b\}^* \text{ et } |w|_a \leq |w|_b\}$,
2. $\{a^n \mid n \text{ est premier}\}$.

Exercice 3 : Opérations sur automates

Soit $L_1 = (a + b)^*ab(a + b)^*a$ et $L_2 = (a + b)b(a + b)^*ab(a + b)^*$.
Construisez l'automate reconnaissant le langage $L_1 \cap mirroir(L_2)$.

Exercice 4 : Une histoire de berger, de loup, de chèvre et de choux

Monsieur Berger B emmène un loup L, une chèvre C, et un choux X près d'une rivière et souhaite traverser avec un petit bateau. Le bateau est tellement petit que B peut entrer dans le bateau avec au plus un passager. Sans surveillance de B, L mange C et C mange X. Nous souhaitons trouver comment B peut faire traverser la rivière à la compagnie sans perdre d'élément.

1. Déterminer l'ensemble des états de l'automate qui représente les différentes situations des deux cotés de la rivière. On pourra associer à chaque état une paire d'ensembles par exemple.
2. Donner un automate qui modélise la situation.
3. Utiliser l'automate trouvé à la question précédente pour trouver comment le problème peut être résolu de manière algorithmique.

Exercice 5 : Chemins dans l'automate

1. Montrer que si un langage fini est reconnu par un automate de n états, alors le langage ne contient pas de mots de longueure supérieur à n.
2. Montrer que si un automate de n états ne reconnait pas de mot de longueur inférieure ou égale à n, alors il ne reconnait aucun mot.

Exercice 6 : Énumérations

On fixe A un automate déterministe sur un alphabet Σ, et un entier $m \in \mathbb{N}^*$.

1. (a) Proposer un algorithme qui calcule la longueur minimale des mots acceptés par A, si cette quantité est bien définie, et qui détecte si elle ne l'est pas.
   (b) On suppose dans cette question que A n'a pas de cycle. Proposer un algorithme qui calcule la longueur maximale des mots acceptés par A, si cette quantité est bien définie, et qui détecte si elle ne l'est pas.
   (c) Que se passe-t-il lorsque A a un cycle ?
2. (a) Proposer un algorithme permettant de calculer le nombre modulo m de mots de longueur n dans $L(A)$.
   (b) Le problème devient-il plus difficile si on demande de calculer le résultat exact (sans modulo) ?
   (c) L'hypothèse que A est déterministe est-elle importante ?

Exercice 7 : Automate à pile

Construisez un automate à pile reconnaissant les langages suivants

— $\{a^n b^n \mid n \in \mathbb{N}\}$,

— $\{w \cdot mirroir(w) \mid w \in \{a, b, c\}^*\}$,

— $\{a^n b^m c^k \mid m + k > n\}$.