Exercice 1 : Automate des résiduels Construire l'automate des résiduels pour les langages suivants

— $\{a, ab, aba, abb\}$

— $ab^* + ba^*$

— $\{w \in \Sigma^* \mid |w|_a - |w|_b \equiv 2 \mod 4\}$

— $\{w \in \Sigma^* \mid |w|_a \equiv 1 \mod 2 \text{ et } |w|_b \equiv 2 \mod 3\}$

Exercice 2 : Automate à pile

Construisez un automate à pile reconnaissant les langages suivants :

— le langage des palindromes,

— les mots contenant autant de a que de b,

— les mots ne contenant pas autant de a que de b,

— le langage $\{a^n b^n \mid n \in \mathbb{N}\} \cup \{a^n b^* c^n \mid n \in \mathbb{N}\}$,

Ces automates sont-ils déterministes ? Peuvent-ils être déterminiser ?

Exercice 3 : Code d'accès Considérons l'alphabet formé par les chiffres utilisés en notation décimale : $\Sigma = \{0, \ldots, 9\}$. Nous souhaitons modéliser des variantes d'automates qui permettent de reconnaitre un code sous la forme $x \cdot y \cdot z$ avec $x, y, z \in \Sigma$. Dans les cas suivants, considérez (quand cela fait sens) les possibilités (1) il existe un bouton pour valider sa saisie et (2) chaque séquence de trois chiffres correspond à une saisie.

1. L'automate ne laisse qu'une seule chance.
2. L'automate ne laisse que deux chances.
3. L'automate ne laisse que deux chances. L'automate permet d'annuler sa saisie grâce à un bouton spécial.
4. L'automate accepte dès que les derniers chiffres entrés correspondent au code.

Exercice 4 : Opérations sur les automates

Lors du dernier TD, nous avions construit des automates pour les langages suivants :

— $L_1 = abb^*a^*$

— $L_2 = a^*b^*$

— $L_3 = ((a+b)^2)^*$

Construisez des automates déterministes complets pour les langages suivants

— $L_5 = L_1 \cdot L_2$

— $L_6 = L_5^*$

— $L_7 = L_6 \cap L_3$

— $L_8 = \overline{L_7}$