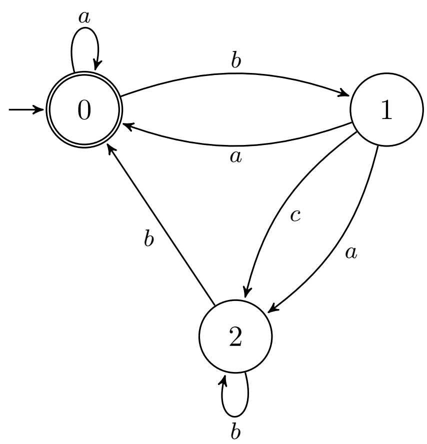
TD 3 : Reconnaissance par morphismes

Exercice 1. (**À préparer** - Automates $\leftrightarrow$ MSO)

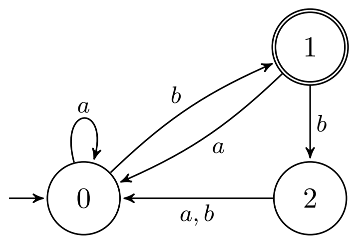
1. Écrire une formule de la logique monadique du second ordre (MSO) pour l'automate ci-dessous.



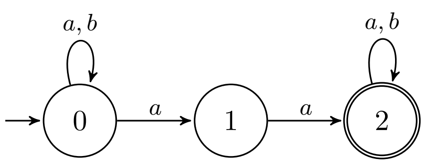
2. Donner des automates équivalents à chacune des formules MSO suivantes :
   (a) $(P_a(x) \rightarrow (x \in Z \leftrightarrow y \in Z)) \vee (P_b(y) \rightarrow (y = x + 2 \wedge P_c(x)))$
   (b) $\exists x\,((P_a(x) \rightarrow (x \in Z \leftrightarrow y \in Z)) \vee (P_b(y) \rightarrow (y = x + 2 \wedge P_c(x))))$
   (c) $\forall x\,((P_a(x) \rightarrow (x \in Z \leftrightarrow y \in Z)) \vee (P_b(y) \rightarrow (y = x + 2 \wedge P_c(x))))$

Exercice 2 (**À préparer** - Monoïdes de transitions)**.**

1. Donner un monoïde fini M et un morphisme φ qui permettent de reconnaître le langage accepté par l'automate suivant :



2. Même question pour l'automate ci-dessous. Attention, l'automate donné n'est pas déterministe. On demande ici de déduire un monoïde de l'automate sans pour autant le déterminiser.



3. En déduire une construction générique d'un monoïde et d'un morphisme reconnaissant le langage d'un automate donné, qu'il soit déterministe ou non.

Exercice 3 (Reconnaissance par monoïde)**.**

1. En utilisant la représentation par monoïdes, montrer que les langages reconnaissables sont clos par union, intersection et complémentaire.
2. En utilisant la représentation par monoïdes, montrer que les langages reconnaissables sont clos par concaténation. (*Indication : si L_1 et L_2 sont reconnus par $\varphi_1 : \Sigma^* \to M_1$ et $\varphi_2 : \Sigma^* \to M_2$ respectivement, on pourra considérer le morphisme $\psi \colon \Sigma^* \to 2^{M_1 \times M_2}$ défini par $\psi(w) = \{(\varphi_1(u), \varphi_2(v)) \mid w = uv\}$*)
3. En déduire que si un langage est sans étoile, alors il est apériodique.
4. En utilisant la représentation par monoïdes, montrer que les langages reconnaissables sont clos par quotient : si L est reconnaissable et $K \subseteq \Sigma^*$, alors $K^{-1}L$ et LK^{-1} sont reconnaissables.

Exercice 4 (Langages sans étoile)**.**

1. Sur l'alphabet $\Sigma = \{a, b\}$, construire les monoïdes syntaxiques des langages suivants : Σ^*, $\Sigma^* a \Sigma^* b \Sigma^*$, $a^* b^*$, $\{w \in \Sigma^* \mid |w|_a \text{ impair}\}$, $((a + cb^*a)c^*b)^*$.
2. Pour chacun de ces langages, justifier s'il est sans étoile ou pas.
3. Pour les langages sans étoile, construire des expressions sans étoile équivalentes, c'est-à-dire en utilisant uniquement des langages finis, l'union, la concaténation, et le complémentaire.
4. Pour les langages sans étoile, construire des formules du premier ordre équivalentes à ces langages.

Exercice 5 (Congruence à droite)**.** Une relation d'équivalence sur Σ^* est dite *congruence à droite* si pour tous u, v, $w \in \Sigma^*$, $u \equiv v$ implique $uw \equiv vw$. On rappelle qu'elle est dite d'*index fini* si elle possède un nombre fini de classes d'équivalence et que le langage L est dit *saturé* par $\equiv$ si pour tout $u \in \Sigma^*$, pour tout $v \in L$, $u \equiv v$ implique $u \in L$.

1. Montrer que $L \subseteq \Sigma^*$ est reconnaissable si, et seulement si, il est saturé par une congruence à droite d'index fini.
2. On définit $\equiv_L^r$ par $u \equiv_L^r v$ si $\forall y \in \Sigma^*$, $uy \in L \iff vy \in L$. Montrer que $\equiv_L^r$ est la congruence à droite la plus grossière qui sature L.
3. Faire le lien entre $\equiv_L^r$ et l'automate minimal de L.

Exercice 6. Soit L un langage reconnaissable. Montrer que le langage

$$L' = \{v \in \Sigma^* \mid v^{|v|} \in L\}$$

est aussi reconnaissable.