TD 4

Exercice 1. *egapmoP*

Un langage vérifie la propriété σ_k pour $k \geq 0$ si, pour toute factorisation $f = uv_1 \dots v_k w$, où tous les v_i sont des mots non-vides, il existe deux indices $1 \leq i < j \leq k$ tels que :

$$f \in L \Leftrightarrow uv_1 \dots v_i v_{j+1} \dots v_k w \in L$$

1. Trouver un langage non régulier qui passe le lemme de l'étoile.
2. Montrer que si L est rationnel, alors il existe k tel que L vérifie σ_k.
3. Montrer que si L vérifie σ_k, alors pour tout mot v, le langage $v^{-1}L$ vérifie σ_k.
4. (Difficile) Montrer par induction sur k qu'il existe un entier $R(k)$ tel que pour tout ensemble S tel que $|S| > R(k)$, et tout coloriage ρ avec deux colours de $\mathcal{P}_2(S)$ (i.e. $\rho : \mathcal{P}_2(S) \to \{0,1\}$ où $\mathcal{P}_2(S)$ est l'ensemble des sous-ensembles de taille 2 de S) il existe un sous-ensemble $T \subset S$ monochromatique de taille k (i.e. ρ est constante sur $\mathcal{P}_2(S)$).
   (Indice : Si on denote $R(a,b)$ le minimum cardinal de S tel que on trouve toujours soit a 0 soit b 1 alors on pourra montrer que $R(a,b) \leq R(a-1,b) + R(a,b-1)$.)
5. On veut montrer qu'il n'existe qu'un nombre fini de langages vérifiant σ_k.
   a) Montrer que si L et K vérifient σ_k et que $L \cap \Sigma^{\leq N} = K \cap \Sigma^{\leq N}$, pour $N = R(k+1)$, alors $L = K$.
      On pourra procéder par induction sur la taille du mot $f = a_0 \dots a_N g$ et considérer l'ensemble $U = \{(i,j) \in [|0;N|]^2 | a_0 \dots a_i a_{j+1} \dots a_N g \in L\}$.
   b) Conclure.
6. Conclure que si un langage vérifie σ_k alors il est rationnel.

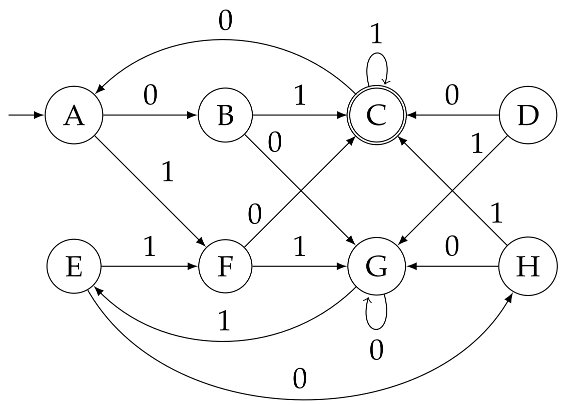
Définition. Soit P une partition. q et q' sont séparables relativement à P s'il existe a et $\delta(q,a)$ et $\delta(q',a)$ qui ne sont pas dans le même ensemble de la partition.

Algorithme pour trouver les paires distingables :
- Éliminer les états non accessibles depuis l'état initial.
- Partition de départ : $F, Q \setminus F$
- Répéter jusqu'à obtenir 2 fois la même partition
  - pour chaque couple d'états (q,q') du même sous-ensemble de la partition, regarder s'ils sont séparables et dans ce cas, rafiner la partition.

Exercice 2. *Automate $\to$ Automate*

1. Écrire un algorithme qui minimise un automate déterministe.

2. Faire tourner l'algorithme sur l'exemple suivant :



3. Montrer que l'algorithme s'arrête (en au plus Q^4 étapes).
4. Montrer que l'automate obtenu est bien équivalent à l'automate initial.
5. Montrer que l'automate obtenu est minimal.
6. Application : Trouver un algorithme pour déterminer si deux expressions rationnelles reconnaissent le meme langage.

Exercice 3. *Trouver de la place*

1. Soit A un langage régulier infini. Prouvez qu'on peut scinder A en deux sous-ensembles réguliers infinis.

Soient B et D deux langages. On note $B \Subset D$ si $B \subseteq D$ et $D \setminus B$ est infini.

2. Montrez que pour tous langages réguliers B et D tels que $B \Subset D$ il existe un langage C régulier tel que $B \Subset C \Subset D$.