TD1 – Ensembles et relations

1. Soit un ensemble E. On définit sur P(E) la relation binaire R : X R Y $\Leftrightarrow$ X et Y ont le même nombre d'éléments. Montrez que R est une relation d'équivalence.

2. Montrez par récurrence le théorème suivant :

Soit A un ensemble fini. $|P(A)| = 2^{|A|}$.

3. Soient R_1 et R_2 deux ordres partiels sur un même ensemble E. Montrez que $R_1 \cap R_2$ est également un ordre partiel.

4. Soient A et B deux ensembles finis. Combien existe-t-il d'applications de A vers B ? Combien sont injectives ? Combien sont surjectives ?

5. Cardinalité. Montrez les propriétés suivantes :

a) L'ensemble des parties d'un ensemble dénombrable n'est pas dénombrable,
b) $\mathbb{N} \setminus \{3,4,5\}$ est dénombrable,
c) S'il existe une injection d'un ensemble E dans $\mathbb{N}$, alors E est fini ou dénombrable,
d) S'il existe une surjection de $\mathbb{N}$ vers un ensemble E, alors E est fini ou dénombrable,
e) Soit E un ensemble dénombrable, et E' $\subset$ E. Alors E' est également dénombrable,
f) $\mathbb{N} \times \mathbb{N}$ est dénombrable,
g) Le produit cartésien de deux ensembles dénombrables est dénombrable,
h) $\mathbb{Q}^+$, l'ensemble des nombres rationnels positifs, est dénombrable.

TD2 – Alphabets, langages, représentation finie

1. Soit Σ un alphabet tel que $|\Sigma| = n$. Combien existe-t-il de mots de longueur $k \geq 0$? Combien existe-t-il de mots de longueur au plus $k \geq 0$?

2. Montrez que pour tout langage L, $L^* = (L^*)^*$.

3. Montrez qu'il existe des langages L_1 et L_2 tels que
- $(L_1 \cup L_2)^* \neq L_1^* \cup L_2^*$
- $(L_1 . L_2)^* \neq L_1^* . L_2^*$

4. (facultatif) Donnez un algorithme pour énumérer tous les mots de longueur au plus k sur un alphabet à n symboles.

> Les **expressions rationnelles** sur un alphabet Σ sont tous les mots construits sur l'alphabet $\Sigma \cup \{(,), \varnothing, \varepsilon, \cup, *\}$:
> (1) $\varnothing$, le mot vide et chaque élément de Σ est une expression rationnelle,
> (2) Si α et β sont des expressions rationnelles, alors $(\alpha\beta)$ et $(\alpha \cup \beta)$ est aussi une expression rationnelle,
> (3) Si α est une expression rationnelle, alors α^* est aussi une expression rationnelle,
> (4) Rien d'autre n'est une expression rationnelle hormis les points (1) à (3).
>
> On dit que deux expressions rationnelles sont équivalentes si elles définissent le même langage.
>
> Soient u et v deux expressions rationnelles. Les égalités suivantes peuvent être démontrées :
>
> | | | |
> |---|---|---|
> | 1. $\varnothing^* = \varnothing$ | 7. $u^*u^* = u^*$ | 13. $(u \cup v)^* = u^*(u \cup v)^*$ |
> | 2. $\varnothing u = u\varnothing = u$ | 8. $(u^*)^* = u^*$ | 14. $(u \cup v)^* = (u \cup vu^*)^*$ |
> | 3. $uu^* = u^*u$ | 9. $u(v \cup w) = uv \cup uw$ | 15. $(u \cup v)^* = (u^*v^*)^*$ |
> | 4. $uu^* \cup \varnothing = u^*$ | 10. $(u \cup v)w = uw \cup vw$ | 16. $(u \cup v)^* = u^*(vu^*)^*$ |
> | 5. $u \cup v = v \cup u$ | 11. $(uv)^*u = u(vu)^*$ | 17. $(u \cup v)^* = (u^*vu^*)^* \cup u^*$ |
> | 6. $u \cup u = u$ | 12. $(u \cup v)^* = (u^* \cup v)^*$ | 18. $(u \cup v)^* = (u^*v)^*u^*$ |

5. Les affirmations suivantes sont-elles vraies ? Expliquez pourquoi :
   a) $baa \in a^*b^*a^*b^*$
   b) $b^*a^* \cap a^*b^* = a^* \cup b^*$
   c) $a^*b^* \cap c^*d^* = \varnothing$
   d) $abcd \in (a(cd)^*b)^*$

6. Soit $\Sigma = \{a, b\}$. Donnez une expression rationnelle définissant les langages sur Σ décrits par les définitions suivantes :
   a) Le langage de tous les mots contenant au moins 2 a (i.e. 2 occurrences de la lettre a),
   b) Le langage de tous les mots contenant au plus 2 a,
   c) Le langage de tous les mots contenant un nombre de a divisible par 3,
   d) Le langage de tous les mots ne contenant pas le facteur aa.

7. Donnez une expression rationnelle définissant le langage $L = \{w \in \{a, b\}^* \mid w \text{ contient aa mais pas abb}\}$
Indication : un mot w du langage peut être décomposée en $w = b^*uv$ où u est une chaîne qui finit par a et ne contient pas abb et v est une chaîne qui commence par a et ne contient pas abb.

8. Pour chacun des langages suivants, donnez une expression rationnelle représentant son complément.
   a) $(a \cup b)^*b$
   b) $((a \cup b)(a \cup b))^*$
   c) $b^*aa^*b(a \cup b)^*$

9. Montrez les égalités suivantes en utilisant les identités ci-dessus :
   a) $((a^*b^*)^*(b^*a^*)^*)^* = (a \cup b)^*$
   b) $(a^*b)^* \cup (b^*a)^* = (a \cup b)^*$
   c) $(a \cup b)^*a(a \cup b)^* = b^*a(a \cup b)^*$

TD 3 – Automates à états finis – Déterminisation

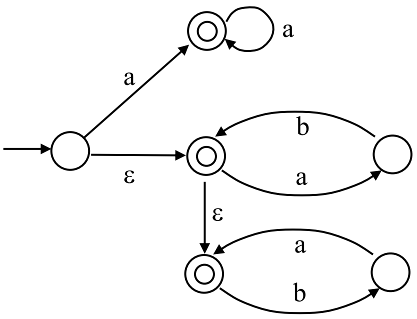
1. On considère l'alphabet $\Sigma = \{a, b\}$. Donnez un exemple d'automate M déterministe, complet ou non, dans les cas suivants :
   a) M n'accepte aucun mot,
   b) M accepte tous les mots sur l'alphabet Σ,
   c) M n'accepte que les mots formés avec une seule lettre de Σ,
   d) M accepte le langage $L = \{a\}^* \cup \{b\}^*$.

2. Construisez un automate déterministe, complet ou non, acceptant les langages suivants :
   a) Les représentations binaires des nombres pairs,
   b) Les représentations décimales des multiples de 3.

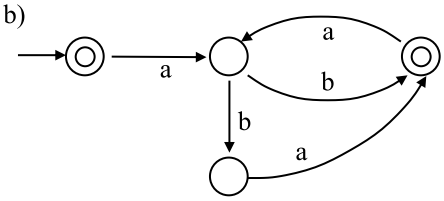
3. Construisez des automates finis déterministes ou non acceptant les langages suivants :
   a) $\{w \in \{a, b\}^* \mid$ chaque a de w est immédiatement précédé et immédiatement suivi par un b$\}$; Notons que le mot babab doit être accepté,
   b) $\{w \in \{a, b\}^* \mid$ w ne contient ni ab ni ba$\}$,
   c) $\{w \in \{a, b\}^* \mid$ w contient ab ou ba $\}$,
   d) $\{w \in \{a, b\}^* \mid$ w contient à la fois ab et ba $\}$. Notez que les mots aba et bab doivent être acceptés,
   e) $\{w \in \{a, b\}^* \mid$ w a un nombre impair de a et un nombre impair de b$\}$,
   f) $\{w \in \{a, b\}^* \mid$ w contient aa et la première occurrence de aa ne doit pas être précédée de abab$\}$,
   g) $\{w \in \{a, b\}^* \mid$ w contient une seule fois la sous-chaîne abba, et la sous-chaîne bb ne figure pas dans w en dehors de abba$\}$,
   h) $\{w \in \{a, b, c\}^* \mid$ w ne contient pas abc$\}$.

4. Quels sont les langages acceptés par chacun des automates suivants ?

a)



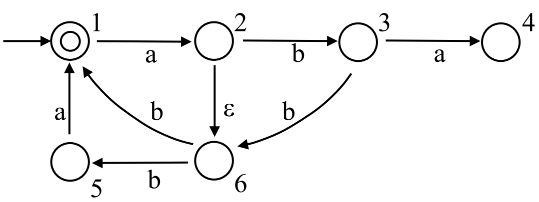
b)



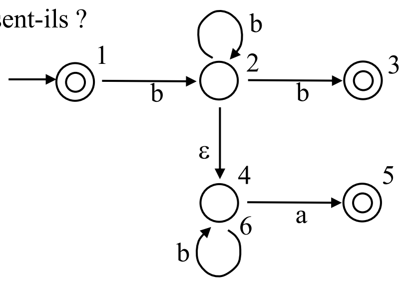
5. Proposez un automate non déterministe acceptant le langage des mots contenant la sous-chaine ba ou bab.

6. Déterminisez les automates suivants. Quels langages reconnaissent-ils ?

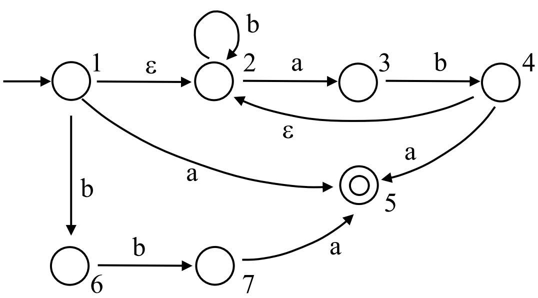
a)



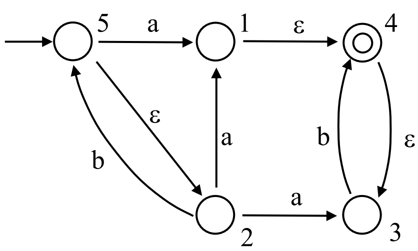
b)



c)



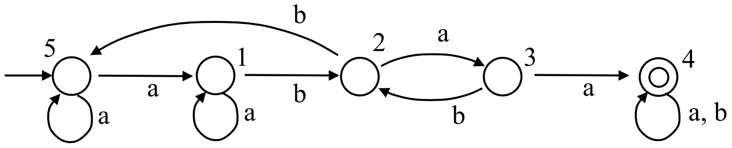
d)



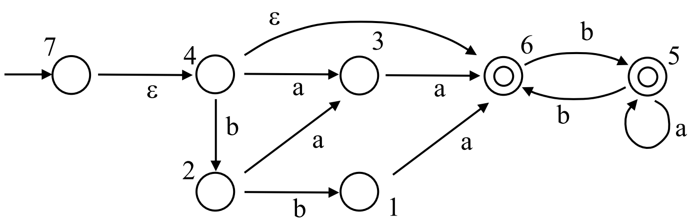
TD 4 – Caractérisation – automate standard

1. Quels langages reconnaissent les automates suivants ?

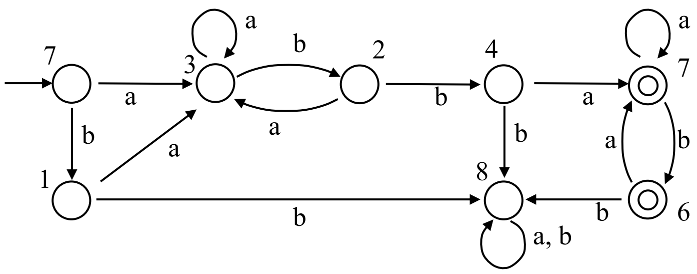
a)



b)



c)



Soit $L \subseteq \Sigma^*$ un langage et $x, y \in \Sigma^*$ deux mots.
On dit que x et y sont équivalents suivant L, et on note $x \approx_L y$, si pour tout mot z de Σ^* :
$xz \in L$ ssi $yz \in L$

Théorème de Myhill – Nerode :
Soit $L \subseteq \Sigma^*$ un langage rationnel.
Il existe un automate déterministe ayant $| \Sigma^* / \approx_L |$ états acceptant L.

Pour un langage L, l'automate standard $M = (K, \Sigma, \delta, s, F)$ est défini de la manière suivante :
$K = \{ [x], x \in \Sigma^* \}$ $F = \{ [x], x \in L \}$
$s = [e]$ δ : définie par $\delta([x], a) = [xa]$

2. Soit le langage $L = \{ w \in \{a, b\}^* \mid |w|_a$ est pair et w ne contient pas bb$\}$.
   a) Calculez les classes d'équivalence suivant L,
   b) Déterminez l'automate standard correspondant à L.

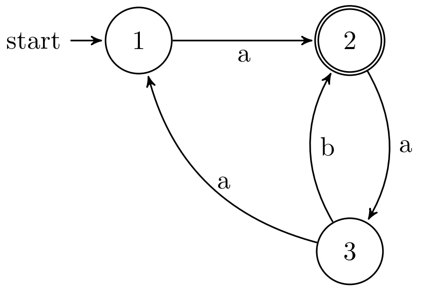
TD 5 – Lemme d'Arden – rationalité

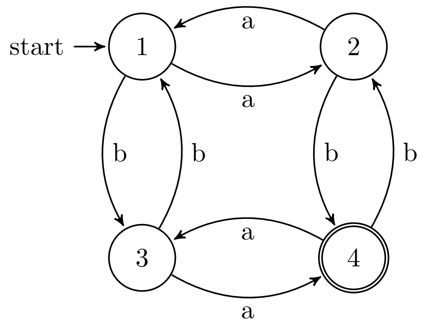
Lemme d'Arden :

$X = e\,X + f$ avec e, f : expressions régulières

$e^*.\,f$ est solution minimale de $X = e\,X + f$ si $\varepsilon \in e$
est solution unique de $X = e\,X + f$ si $\varepsilon \notin e$

1. En utilisant le lemme d'Arden, donnez une expression régulière décrivant le langage reconnu par les automates :





La classe des langages rationnels est stable pour l'union, la concaténation, l'étoile de Kleene, l'intersection et le complémentaire.

Pour montrer qu'un langage L est **non** rationnel, on fait le raisonnement par l'absurde suivant :

- Supposer L rationnel ;
- Déterminer L_1 rationnel tel que L_1 op L = L_2 (op $\in$ {$\cup$, ., *, $\cap$, complémentaire}), L_2 connu non rationnel ;
- L_1 op L rationnel par stabilité, L_2 connu non rationnel, donc contradiction.

2. Montrez que le complémentaire d'un langage non rationnel est non rationnel.

3. Est-ce que la classe des langages non rationnels est stable par
   a) Union ?
   b) Intersection ?
   c) Étoile de Kleene ?

4. À l'aide des propriétés de stabilité, montrez que le langage L = { w $\in$ {a,b}* | w comporte autant de a que de b } est non rationnel.

Lemme de l'étoile :

Soit L un langage rationnel infini accepté par un automate <u>déterministe</u> M à k états.
Soit z un mot quelconque de L tel que $|z| \geq k$.
Alors z peut être décomposé en uvw avec $|uv| \leq k$, $|v| \neq 0$ et $uv^iw \in L$, $\forall\, i \geq 0$.

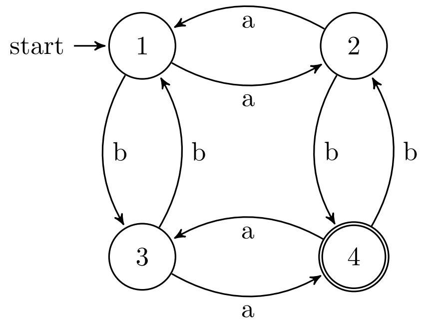
5. À l'aide du lemme de l'étoile, montrez que le langage L = $\{a^pb^q \mid p \in \mathbf{N}, q \in \mathbf{N}, q \leq p\}$ est non rationnel.

TD 6 – Grammaires

1. Soit $\Sigma = \{a, b\}$. Construisez des grammaires correspondant aux langages :
   a) a^*
   b) $(a \cup b)(a \cup b)^*$
   c) $(a \cup b)^*$
   d) $(a \cup b)^* aa (a \cup b)^*$

2. Soit $\Sigma = \{a, b\}$. Soit le langage $(a \cup b)^* aa (a \cup b)^*$.
   a) Construisez une grammaire régulière correspondant à ce langage,
   b) Déduisez l'automate à états finis de la grammaire construite.

3. Construisez la grammaire régulière engendrant le langage reconnu par l'automate suivant.



4. Construisez l'automate à pile acceptant le langage $\{a^{2n} b^n \mid n \geq 0\}$.

5. Soit la grammaire LR(1)
   $E \rightarrow E \vee T \mid T$
   $T \rightarrow T \wedge F \mid F$
   $F \rightarrow \neg F \mid (E) \mid 0 \mid 1$
   a) Effectuez l'analyse ascendante de l'expression $0 \vee 1$ puis de l'expression $(0 \vee \neg 0) \wedge (\neg 1 \vee 0)$,
   b) Ajoutez des actions à la grammaire pour évaluer une expression,
   c) Évaluez les expressions précédentes.

6. Soit la grammaire LR(1)
   I → *if* B *then* I | *if* B *then* I *else* I | *o*
   B → *b*
   a) Montrez que cette grammaire est ambiguë,
   b) Effectuez l'analyse ascendante de l'instruction *if b then if b then o else o* ; À quelle étape a-t-on le choix entre une lecture et une réduction ? Donnez les arbres de dérivation dans chacun des cas,
   c) Construisez une grammaire non ambiguë équivalente qui associe chaque *else* avec le *then* le plus proche qui ne correspond pas déjà à un *else*.