POLYTECHNIQUE MONTRÉAL
UNIVERSITÉ D'INGÉNIERIE

LOG1810
STRUCTURES DISCRÈTES

TD 13 : MODÉLISATION COMPUTATIONNELLE

A2023

SOLUTIONNAIRE

Exercice 1

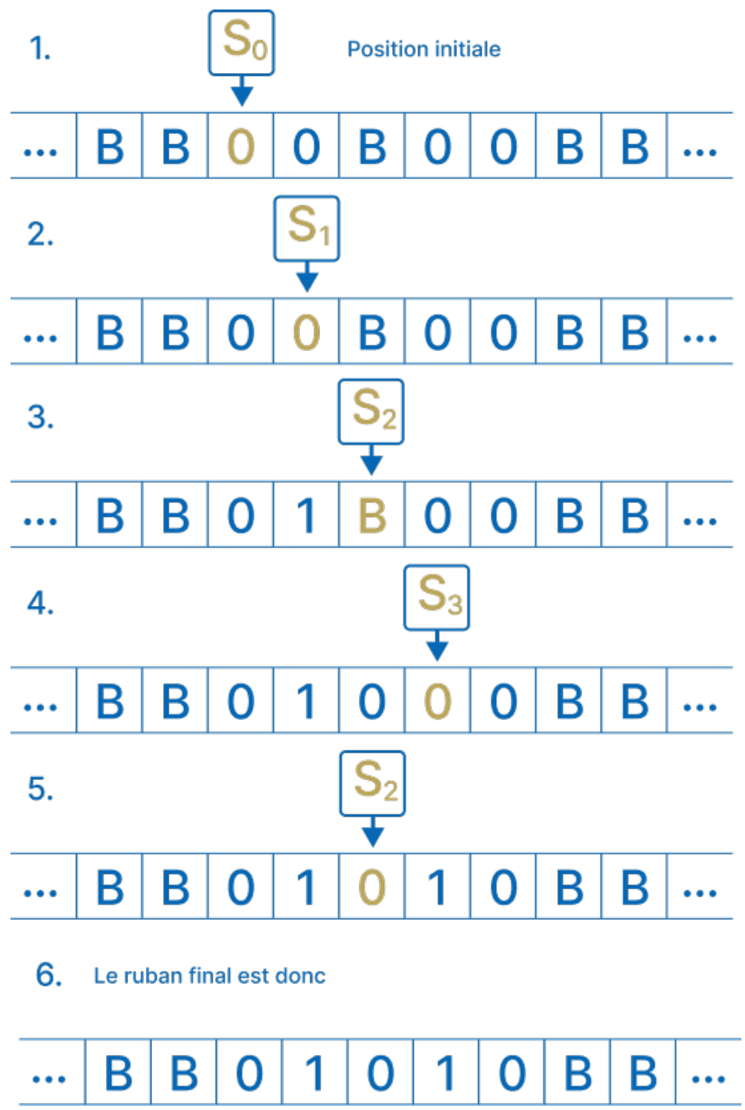
Soit M_T la machine de Turing dont l'état initial est S_0 et définie par les huit quintuples suivants :

- $(S_0, 0, S_1, 0, R)$
- $(S_0, 1, S_1, 0, L)$
- $(S_0, B, S_1, 1, R)$
- $(S_1, 0, S_2, 1, R)$
- $(S_1, 1, S_1, 1, R)$
- $(S_1, B, S_2, 0, R)$
- $(S_2, B, S_3, 0, R)$
- $(S_3, 0, S_2, 1, L)$

En considérant le ruban initial donné, déterminez le ruban final lorsque M_T s'arrête. On suppose que M_T commence en position initiale.

…	B	B	0	0	B	0	0	B	B	…

Solution :



$(S_0, 0, S_1, 0, R)$

$(S_1, 0, S_2, 1, R)$

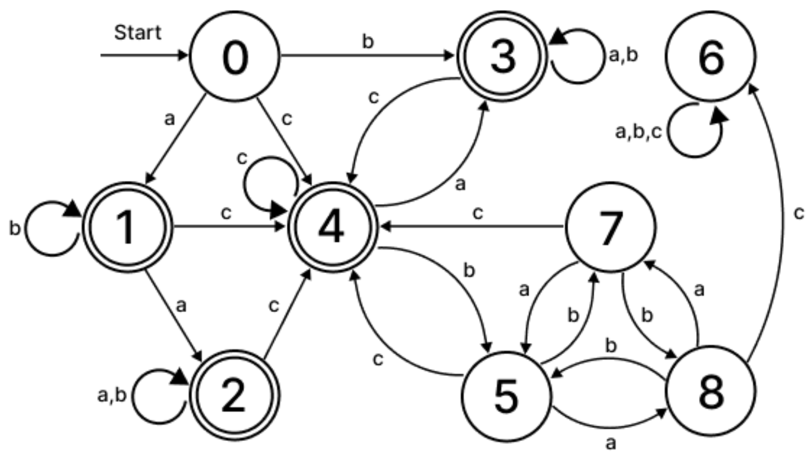
$(S_2, B, S_3, 0, R)$

$(S_3, 0, S_2, 1, L)$

Finalement, à la 5e étape, M_T s'arrête car il n'y a aucun quintuple commençant par la paire $(S_2, 0)$ dans la description de la machine.

Exercice 2

Donnez la grammaire G qui génère le langage reconnu par l'automate suivant. Vous devez préciser l'alphabet V, l'ensemble des symboles terminaux T, l'axiome S et l'ensemble des règles de production P.



Solution :

Soit les symboles non terminaux associés aux états comme suit :

- État 0 : Symbole non terminal S, axiome de la grammaire
- État 1 : Symbole non terminal A
- État 2 : Symbole non terminal B
- État 3 : Symbole non terminal C
- État 4 : Symbole non terminal D
- État 5 : Symbole non terminal E
- État 7 : Symbole non terminal F
- État 8 : Symbole non terminal H

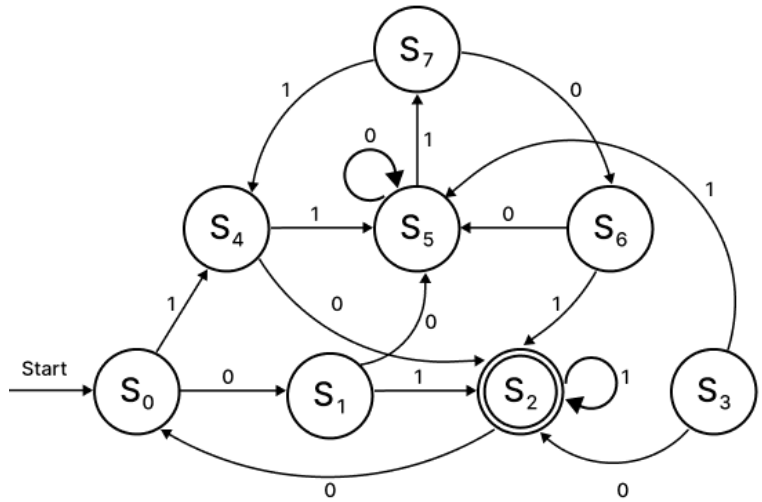
Nous avons les ensembles suivants :

$V = \{\mathrm{a}, \mathrm{b}, c, S, A, B, C, D, E, F, H\}$
$T = \{\mathrm{a}, \mathrm{b}, \mathrm{c}\}$
Les productions de P sont :

$S \rightarrow \mathrm{aA} \mid \mathrm{bC} \mid \mathrm{cD} \mid \mathrm{a} \mid b \mid c$
$A \rightarrow aB \mid bA \mid cD \mid a \mid b \mid c$
$B \rightarrow aB \mid bB \mid cD \mid a \mid b \mid c$
$C \rightarrow aC \mid bC \mid cD \mid a \mid b \mid c$
$D \rightarrow aC \mid bE \mid cD \mid a \mid c$
$E \rightarrow aH \mid bF \mid cD \mid c$
$F \rightarrow aE \mid bH \mid cD \mid c$
$H \rightarrow aF \mid bE$

Exercice 3

Minimisez l'automate ci-dessous. Donnez la table d'états-transition et précisez les états finaux. Présentez toutes les étapes de votre démarche. Enfin, construisez l'automate que vous proposez.



Solution :

1. $A = \{S_2\}$, $B = \{S_0, S_1, S_3, S_4, S_5, S_6, S_7\}$

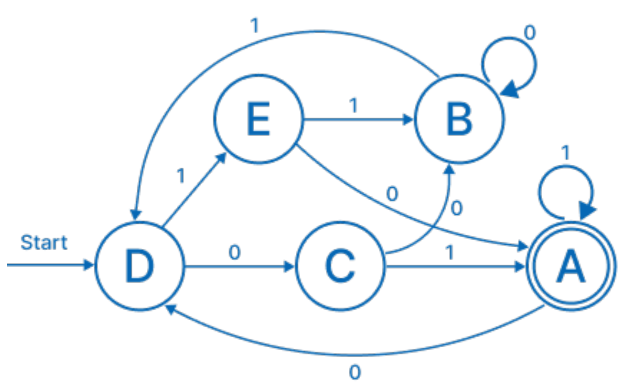
2. $A = \{S_2\}$, $B = \{S_0, S_5, S_7\}$, $C = \{S_1, S_3, S_4, S_6\}$

3. $A = \{S_2\}$, $B = \{S_5\}$, $C = \{S_1, S_6\}$, $D = \{S_0, S_7\}$, $E = \{S_3, S_4\}$

La table d'états-transition de l'automate minimisé est la suivante. Les états initiaux et finaux sont marqués des signes $\longrightarrow$ et $\longleftarrow$, respectivement.

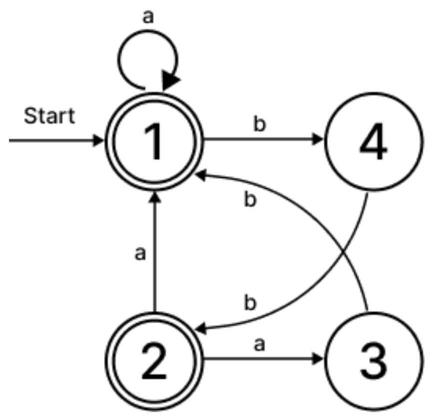
États	**Entrée**	
	0	**1**
$\longleftarrow A$	D	A
B	B	D
C	B	A
$\longrightarrow D$	C	E
E	A	B

L'automate est :



Exercice 4

En utilisant le lemme d'Arden, trouvez le langage reconnu par la machine à états finis suivante. Présentez toutes les étapes de votre démarche.



Solution :

Soient X_1, X_2, X_3 et X_4 les étiquettes associées aux états 1, 2, 3 et 4, respectivement.

Le système d'équations décrivant les états de l'automate est :

$$\begin{cases} X_1 = aX_1 + bX_4 + \epsilon \\ X_2 = aX_1 + aX_3 + \epsilon \\ X_3 = bX_1 \\ X_4 = bX_2 \end{cases}$$

En substituant X_3 dans X_2, on a :

$$\begin{aligned} X_2 &= aX_1 + a(bX_1) + \epsilon \\ &= aX_1 + a(bX_1) + \epsilon \\ &= (a + ab)X_1 + \epsilon \end{aligned}$$

En substituant le résultat obtenu pour X_2 dans X_4, on obtient :

$$\begin{aligned} X_4 &= b\big((a + ab)X_1 + \epsilon\big) \\ &= (ba + bab)X_1 + b \end{aligned}$$

En substituant le résultat obtenu pour X_4 dans X_1, on obtient :

$$\begin{aligned} X_1 &= aX_1 + b\big((ba + bab)X_1 + b\big) + \epsilon \\ &= (a + bba + bbab)X_1 + bb + \epsilon \\ &= (a + bba + bbab)^*(bb + \epsilon) && lemme\ d'Arden \end{aligned}$$

Le langage reconnu par cette machine à état est donc $(a + bba + bbab)^*(bb + \epsilon)$.

Exercice 5

Montrez que le langage $\boldsymbol{L}$ n'est pas régulier.

$$\boldsymbol{L} = \{0^n \mid n\ est\ un\ nombre\ premier\}$$

Solution :

Raisonnons par l'absurde.
Supposons que le langage $\boldsymbol{L}$ est régulier. Il vérifie donc le lemme de pompage.
Soit p un nombre premier supérieur au seuil de pompage.

Considérons le mot $w = 0^p$, qui appartient à $\boldsymbol{L}$.
Par le lemme de pompage, il existe une décomposition $w = xyz$ tel que $x = 0^q, y = 0^r$ et $z = 0^{p-q-r}$ avec $q + r \leq p$ (car $|xy| \leq p$) et $r > 0$ (car $y \neq \varepsilon$).

Selon le lemme de pompage, $\forall i \geq 0, xy^iz \in \boldsymbol{L}$.
Ainsi, le mot $xy^{(p+1)}z$ devrait également être dans $\boldsymbol{L}$.

Pour $i = p + 1$, nous avons : $xy^iz = 0^q(0^r)^{(p+1)}0^{p-q-r}$
$= 0^q0^{r(p+1)}0^{p-q-r}$
$= 0^{p+r(p+1)-r}$
Simplifiant, cela donne $0^{p(1+r)}$

Cependant, $p(1 + r)$ n'est pas un nombre premier, puisque $r > 0$.

Par conséquent, lorsque $i = p + 1$, $xy^iz \notin \boldsymbol{L}$. Le lemme de pompage n'est pas vérifié.
D'où le langage $\boldsymbol{L}$ n'est pas régulier.
CQFD

Exercice 6

On considère :

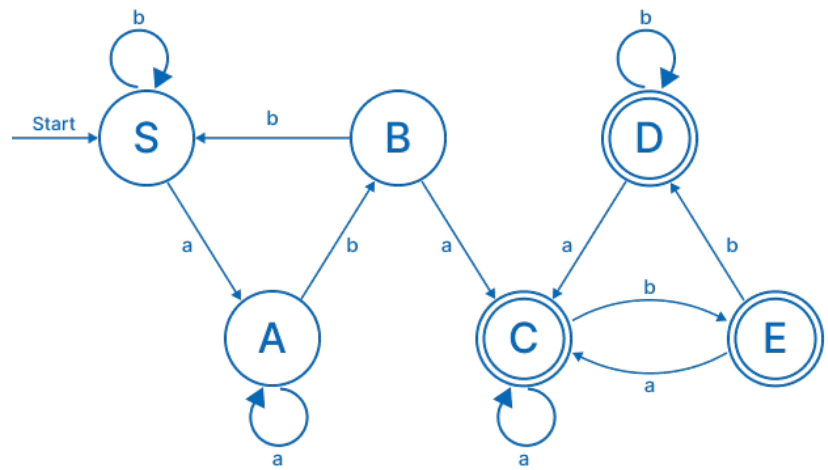
- Le vocabulaire $V = \{a, b, S, A, B, C, D, E\}$
- L'ensemble des symboles terminaux $T = \{a, b\}$
- L'axiome S

a) Soit la grammaire $G_1 = (V, T, S, P_1)$ avec P_1 l'ensemble des règles de production suivant :

$$S \to aA \mid bS$$
$$A \to aA \mid bB$$
$$B \to aC \mid bS \mid a$$
$$C \to aC \mid bE \mid a \mid b$$
$$D \to aC \mid bD \mid a \mid b$$
$$E \to aC \mid bD \mid a \mid b$$

Construisez l'automate M_1 tel que $L(G_1) = L(M_1)$.

Solution :

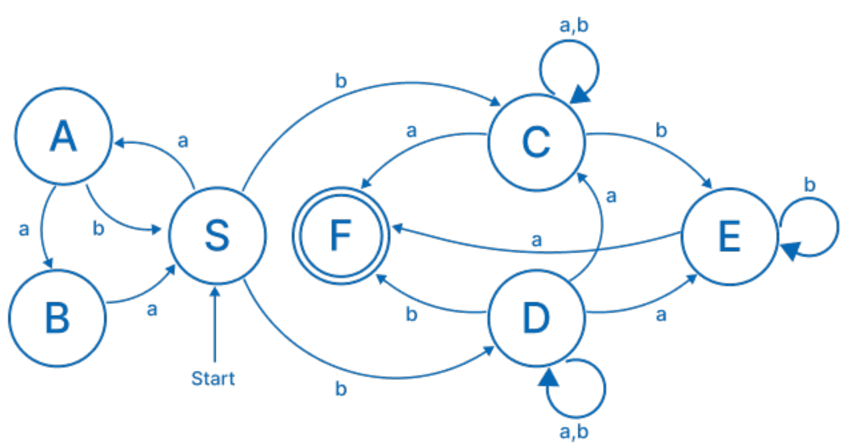


b) Soit la grammaire $G_2 = (V, T, S, P_2)$ avec P_2 l'ensemble des règles de production suivant :

$$
\begin{aligned}
&S \to aA \mid bC \mid bD \\
&A \to aB \mid bS \\
&B \to aS \\
&C \to aC \mid bC \mid bE \mid a \\
&D \to aC \mid aD \mid aE \mid bD \mid b \\
&E \to bE \mid a
\end{aligned}
$$

Construisez l'automate M_2 tel que $L(G_2) = L(M_2)$.

Solution :



Exercice 7

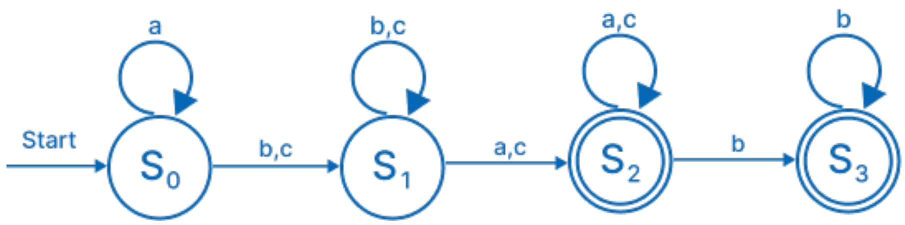
Considérez l'ensemble des symboles terminaux $I = \{a,\ b,\ c\}$.

a) Construisez un automate fini reconnaissant l'expression :

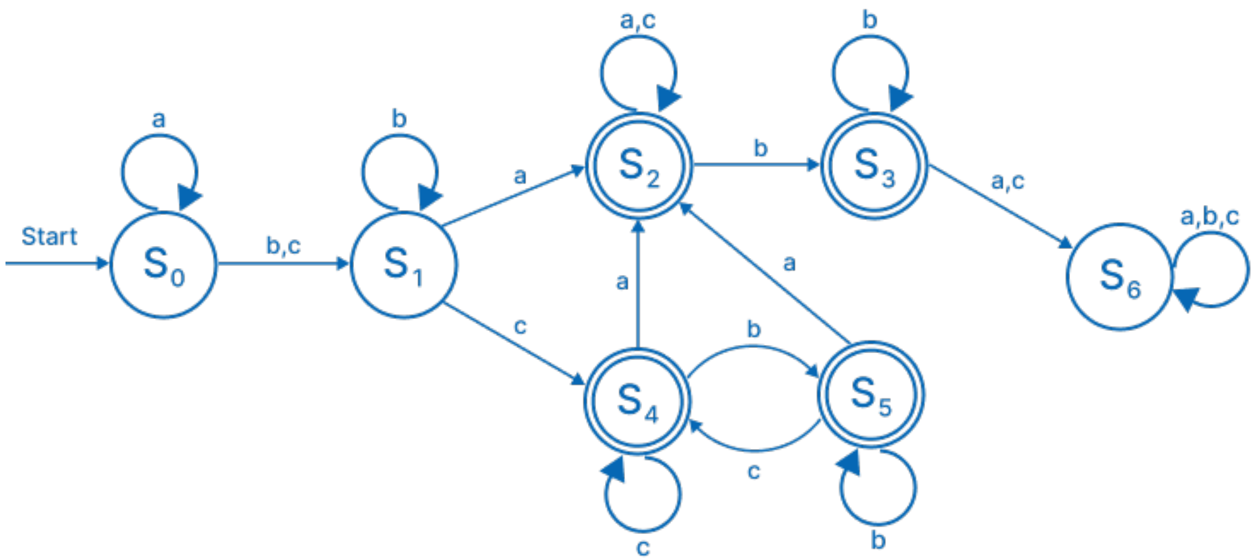
$$a^*(b + c)^+(a + c)^+b^*$$

Solution :

- S_0 est l'état initial de l'automate.
- S_2 et S_3 sont les états d'acceptation, finaux ou terminaux de l'automate.
- Automate fini non déterministe



- Automate fini déterministe

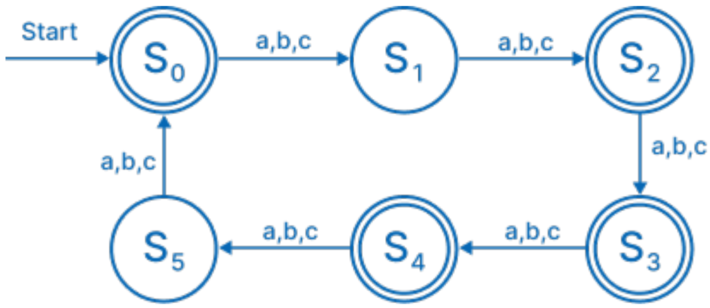


b) Construisez un automate fini déterministe à **6 états** reconnaissant l'expression :

$$((a+b+c)^2)^* + ((a+b+c)^3)^*$$

Solution :

- S_0 est l'état initial de l'automate.
- S_0, S_2, S_3 et S_4 sont les états d'acceptation, finaux ou terminaux de l'automate.
- Automate



Exercice 8
Construisez une machine de Turing qui reconnaît l'ensemble de toutes les chaînes de bits qui contiennent au moins deux « 1 ». Justifiez votre réponse.

Solution :

Nous pouvons rester dans S_0 jusqu'à ce que nous atteignions le premier « 1 » et puis rester dans l'état S_1 jusqu'à ce que nous atteignions le deuxième « 1 ». À ce stade, nous pouvons entrer dans l'état S_2 qui sera un état d'acceptation.

Si nous arrivons au dernier blanc alors que nous sommes toujours dans les états S_0 ou S_1, nous n'accepterons pas cette chaîne de bits.

Ainsi, les quintuples sont donc :

- $(S_0, 0, S_0, 0, R)$
- $(S_0, 1, S_1, 1, R)$
- $(S_1, 0, S_1, 0, R)$
- $(S_1, 1, S_2, 1, R)$

Exercice 9 (Facultatif)
Montrez que le langage $\boldsymbol{L}$ n'est pas régulier.

$$\boldsymbol{L} = \{0^{n!} \mid n \in \mathbb{N}\}$$

Solution :

Raisonnons par l'absurde.
Supposons que le langage $\boldsymbol{L}$ est régulier. Il vérifie donc le lemme de pompage.
Soit p le seuil de pompage.
Le mot $w = 0^{p!}$ est un mot de $\boldsymbol{L}$ (sauf si $p < 3$, auquel cas nous choisirons $0^{3!}$).
Il existe une décomposition $w = xyz$ tel que $x = 0^q, y = 0^r$ et $z = 0^{p!-q-r}$
avec $q + r \leq p$ (car $|xy| \leq p$) et $r > 0$ (car $y \neq \varepsilon$).

D'après le lemme de pompage, $\forall i \geq 0, xy^iz \in \boldsymbol{L}$.
Ainsi, le mot xy^0z devrait être aussi un mot de $\boldsymbol{L}$.
Pour $i = 0$, on a :

$$\begin{aligned} xy^iz &= 0^q(0^r)^0 0^{p!-q-r} \\ &= 0^q 0^{p!-q-r} \\ &= 0^{p!-r} \end{aligned}$$

Pour que $0^{p!-r}$ soit un mot de $\boldsymbol{L}$, il doit y avoir un entier s tel que $s! = p! - r$.
Cependant, cela n'est pas possible puisque lorsque $p \geq 3$ et $r \leq p$, on a :

$$p! - p \leq p! - r \qquad \textbf{(I)}$$

Or, $p! - p = p \cdot (p-1)! - p = p\big((p-1)! - 1\big)$

Et $(p-1)! < p\big((p-1)!-1\big)$, $car\ p \geq 3$

Soit

$$(p-1)! < p! - p \qquad \textbf{(II)}$$

Par $(\mathbf{I})$ et $(\mathbf{II})$, on obtient :

$$(p-1)! < p! - p < p! - r$$

Soit

$$(p-1)! < p! - r \qquad \textbf{(III)}$$

Aussi, lorsque $p \geq 3$ et $r \leq p$, on a :

$$p! - r < p! \qquad \textbf{(IV)}$$

Avec $(\mathbf{III})$ et $(\mathbf{IV})$, on obtient :

$$(p-1)! < p! - r < p!$$

Ainsi, on en déduit que $p! - r$ ne peut être factoriel d'un entier.

Donc lorsque $i = 0$, $xy^iz \notin \boldsymbol{L}$. Le lemme de pompage n'est pas vérifié.
D'où le langage $\boldsymbol{L}$ n'est pas régulier.
CQFD