DLST UNIVERSITÉ Grenoble Alpes

Examen final du 9/01/2020
Licence Sciences et Technologies, 2ème année

INF 302 : Langages et Automates
Année académique 2019/2020

Lire complètement les consignes avant de répondre à l'examen.

Consignes et informations générales

— Durée : 2 heures (13h → 15h).
— Aucune sortie avant 30 minutes.
— Aucune entrée après 30 minutes.
— Matériel nécessaire : stylo à encre noire.
— Matériel conseillé : blanc correcteur (tipex), crayon à papier et gomme.
— 5 feuilles A4 R/V autorisées.
— Tout dispositif électronique est interdit (calculatrice, téléphone, tablette, etc.).
— Le soin de la copie sera pris en compte.
— Le barème est donné à titre indicatif.

Consignes et informations en rapport avec le QCM

— **Les réponses aux questions sont à donner exclusivement sur les feuilles de réponses : les réponses données ailleurs seront ignorées.**
— **Vous devez rendre 1) une copie double de type examen sans aucune inscription (à l'exception de vos informations d'identification) 2) et la feuille de réponse.**
— **Les réponses finales sont à indiquer avec un stylo à encre noire.** Ne pas utiliser de feutre.
— Sauf mention contraire dans l'énoncé, répondre à une question consiste à marquer **toutes les cases** correspondant aux affirmations que vous pensez être correctes ou à indiquer votre réponse à la question (exclusivement) dans le champ texte prévu à cet effet (si celui-ci est présent).
— Pour marquer une case, il faut **colorier entièrement** les cases. Ne pas cocher, mettre de croix ou de signe dans la case. Voir Figure 1. Colorier avec un stylo **noir**. Conseil : commencer par marquer vos réponses avec un crayon à papier puis colorier au stylo noir avant la fin de l'examen. Si vous souhaitez annuler un choix, mettre du Tipex sur la case (pas besoin de redessiner la case).
— Marquer une case se rapportant à une affirmation correcte donne des points, marquer une case se rapportant à une affirmation incorrecte enlève des points, ne pas marquer de cases n'a pas d'influence sur les points accumulés.
— Les questions faisant apparaître le symbole ♣ peuvent présenter une ou plusieurs affirmations correctes. Les autres ont une unique bonne réponse (une seule case à cocher).
— Pour les questions avec une unique bonne réponse, cocher plusieurs cases annule la réponse.
— Dans les feuilles de réponse, ne rien inscrire dans les cases réservées aux enseignants (avec indication *Réservé enseignant*). Toute inscription dans cette case entraine la nullité de la réponse à la question.
— Les parties sont indépendantes. Il est conseillé de lire toutes les questions dans une partie avant de commencer à répondre à cette partie.

(-i-) KO (-ii-) KO (-iii-) KO (-iv-) OK

FIGURE 1 – Comment marquer une case.

— **Attention, certaines questions peuvent être coupées entre deux pages.**

Rappels et notations. Pour un ensemble E, nous notons $|E|$ le cardinal de E. Les symboles $\subseteq$ et $\subset$ dénotent les relations d'inclusion et d'inclusion stricte entre ensembles.

La longueur d'un mot u est dénotée par $|u|$. Le symbole $\cdot$ dénote l'opérateur de concaténation entre mots ou entre langages selon le contexte.

Pour deux mots w et w' sur un alphabet Σ,

— w est un préfixe de w', noté $w \preceq w'$, s'il existe un mot w'' tel que $w' = w \cdot w''$.

— w est une extension de w' si w' est un préfixe de w.

La fermeture par une relation d'un langage est le plus petit ensemble contenant les éléments de ce langage et les éléments en relation avec les éléments de ce langage. Par exemple, la fermeture par préfixe d'un langage L est l'ensemble contenant L et tous les préfixes des mots de L et la fermeture par extension d'un langage L est l'ensemble contenant L et toutes les extensions des mots de L. Un ensemble est clos par une relation s'il est égal à sa fermeture par cette opération.

Un AEFD est un automate à états fini et déterministe. Un AEFND est un automate à états fini et non-déterministe. Un ϵ-AEFND est un automate à états fini avec ϵ-transitions et non-déterministe. Pour un automate quelconque, nous notons $L(A)$ le langage reconnu par A. Le langage d'une expression régulière r est dénoté par $L(r)$.

On dit qu'un automate est équivalent à une expression régulière si le langage reconnu par l'automate est égal au langage dénoté par l'expression régulière.

Un problème ou une question est décidable s'il existe un algorithme qui permet de répondre à la question. Dans le cas contraire, on dit que le problème est indécidable.

Un entier n est une constante d'itération d'un langage régulier si chaque mot de longueur supérieure ou égale à n peut être décomposé comme décrit par le lemme de l'itération. La constante d'itération minimale d'un langage est sa plus petite constante d'itération.

Sujet

Champ Libre

Question 1 [`champ-libre`] Vous pouvez utiliser l'espace de texte de cette question comme champ libre où vous pouvez ajouter toute information concernant l'examen que vous jugerez utile.

Partie 1 : Questions générales (3,5 points)

Question 2 [`cours-1`] ♣ **(0,5 points)** Un automate non-déterministe avec ϵ-transitions qui reconnaît le langage universel...

[redacted] peut avoir un seul état accepteur [redacted] peut avoir plusieurs états accepteurs
[redacted] peut avoir des états non-accepteurs d est forcément déterministe
e a forcément aucun état non-accepteur f n'a pas d'expression régulière équivalente
g est minimal h a un nombre infini d'états
i Les affirmations données dans les réponses entre a et h sont toutes correctes.
j Aucune des affirmations données dans les réponses entre a et h n'est correcte.

Question 3 [`cours-2`] ♣ **(0,75 point)** Soit L un langage pour lequel il n'existe pas d'expression régulière qui le dénote...

[redacted] L ne peut pas être reconnu par un automate [redacted] L n'est pas un langage à état
[redacted] L est de cardinal infini [redacted] L est non-régulier [redacted] $\overline{L}$ est non-régulier
[redacted] $L \cup \Sigma^*$ satisfait le lemme de l'itération g L est régulier h $\overline{L}$ est régulier
i L satisfait forcément le lemme de l'itération
j $\overline{L}$ satisfait forcément le lemme de l'itération
k $\overline{L}$ ne peut pas satisfaire le lemme de l'itération
l L ne peut pas satisfaire le lemme de l'itération
m $L \cup \Sigma^*$ ne peut pas satisfaire le lemme de l'itération
n Les affirmations données dans les réponses entre a et m sont toutes correctes.
o Aucune des affirmations données dans les réponses entre a et m n'est correcte.

Question 4 [cours-3] ♣ **(0,5 points)** Soient L_1 et L_2 deux langages réguliers sur un alphabet Σ.

[redacted] Déterminer si $L_1 \cap L_2$ est de cardinal fini est *décidable.*
[redacted] Déterminer si $L_1 \cup L_2$ est de cardinal fini est *décidable.*
[redacted] $L_1 \cap L_2$ est toujours un langage régulier. [redacted] $L_1 \cup L_2$ est toujours un langage régulier.
e Déterminer si $L_1 \cap L_2$ est de cardinal fini est *indécidable.*
f Déterminer si $L_1 \cup L_2$ est de cardinal fini est *indécidable.*
g Il manque des données pour déterminer si les précédentes affirmations sont correctes ou non.

Question 5 [cours-5] **(0,25 points)** Soient L_1 et L_2 deux langages non réguliers sur un alphabet Σ tels que $L_1 \subseteq L_2$.

a $L_1 \cup L_2$ est un langage régulier. [redacted] $L_1 \cup L_2$ est un langage non régulier.
c Il manque des données pour déterminer si $L_1 \cup L_2$ est régulier ou non.

Question 6 [cours-6] ♣ **(0.5 points)** Soient e et f deux expressions régulières.

[redacted] $L(e+f) = L(f+e)$.
[redacted] $L((e+\epsilon)^*) = L(e^*)$.
[redacted] $L((e+f)^*) = L((f+e)^*)$.
[redacted] $L(e+\emptyset) = L(\emptyset+e) = L(e)$.
[redacted] si $\epsilon \in L(e)$, alors $L(e^+) = L(e^*)$.
f $L((e+f)^*) = L(e^*) + L(f^*)$.
g $L(e\cdot\emptyset) = L(\emptyset\cdot e) = L(e)$.
h $L(e\cdot f) = L(f\cdot e)$.
i $L((e+\epsilon)\cdot e^+) = L(e^*)$.
j $L((e\cdot f)^*) = L((e+f)^*)$.
k $L((e+f)^*\cdot e) = L((e^*\cdot f)^*)$.
l Aucune des affirmations données dans les réponses entre a et k n'est correcte.

Question 7 [cours-7] **(0,25 points)**
On applique l'algorithme de minimisation vu en cours sur un automate. Après trois étapes de l'algorithme, on obtient le résultat ci-contre.

$\equiv_0$	$\equiv_1$	$\equiv_2$
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5

a Cet automate est minimal.
b Cet automate est non minimal.
[redacted] Il manque des données pour déterminer si l'automate est minimal.

Question 8 [cours-9] ♣ **(0,75 point)** Considérons un automate A sur l'alphabet Σ, son automate complété $C(A)$ et son automate complémentaire A^C, obtenus suivant les procédures vues en cours.

■ $L(A) = L(C(A))$. b $L(A) = L(A^C)$. c $\Sigma^* \setminus L(A) = L(C(A))$.
d $L(A) \setminus \Sigma^* = L(A^C)$. ■ $\Sigma^* \setminus L(A) = L(A^C)$. f $L(A) \setminus \Sigma^* = L(A^C)$.
g Aucune des affirmations données dans les réponses entre a et f n'est correcte.
h Les affirmations données dans les réponses entre a et f sont toutes correctes.

Partie 2 : Déterminisation d'ϵ-AEFND (3 points)

Question 9 [determinisation] **(3 points)** Considérons l'ϵ-AEFND représenté dans la Figure 2-i-. Le déterminisé de cet automate est celui représenté dans

■ la Figure 3-iv-. b la Figure 3-i-. c la Figure 3-ii-. d la Figure 3-iii-.
e Aucune des figures.

Partie 3 : Automate vers expression régulière (1,5 points)

Nous considérons l'automate dans la Figure 2-ii-. Les états sont numérotés de 1 à 3.

Question 10 [auto2er-1] ♣ **(1,5 point)** Nous utilisons la méthode associant des équations aux chemins. Calculer les $R^0_{i,j}$. N ous obtenons :

■ $R^0_{11} = \epsilon + b$ ■ $R^0_{12} = a$ ■ $R^0_{21} = b$ ■ $R^0_{22} = \epsilon$ ■ $R^0_{23} = a$.
■ $R^0_{31} = \emptyset$ ■ $R^0_{32} = b$ ■ $R^0_{33} = a + \epsilon$
i $R^0_{11} = b$ j $R^0_{12} = b$ k $R^0_{21} = a$ l $R^0_{22} = \epsilon + b$ m $R^0_{23} = a + \epsilon$.
n $R^0_{31} = b$ o $R^0_{32} = a$ p $R^0_{33} = b + \epsilon$
q Aucune des propositions de $R^0_{i,j}$.

Partie 3bis : Automate vers expression régulière (3 points)

Nous considérons l'automate dans la Figure 2-ii-. Nous souhaitons trouver l'expression régulière associée à cet automate, c'est à dire l'expression régulière qui dénote le langage accepté par cet automate. Pour cela, nous utilisons la méthode associant des équations aux états. Les états sont numérotés de 0 à 3 et X_i dénote le langage accepté à partir de l'état numéro i, pour i entre 0 et 3. Appliquer la méthode en suivant les consignes données dans les questions (dans l'ordre).

Question 11 [auto2erbis-1] ♣ **(0,75 point)** Écrire le système d'équations associé à cet automate. Ensuite, indiquer les équations correctes parmi les suivantes. Il y a au plus une équation correcte par état.

■ $X_1 = aX_2 + bX_1$ ■ $X_2 = aX_3 + bX_1$ ■ $X_3 = aX_3 + bX_2 + \epsilon$
d $X_1 = aX_2 + bX_1 + \epsilon$ e $X_1 = aX_1 + bX_2 + \epsilon$ f $X_1 = X_2b + X_1a$
g $X_1 = X_2a + X_1b + \epsilon$

[h] $X_2 = aX_1 + bX_3$ [i] $X_2 = aX_1 + bX_3 + \epsilon$ [j] $X_2 = aX_3 + bX_1 + \epsilon$
[k] $X_2 = aX_3 + bX_1 + b$
[l] $X_3 = aX_3 + bX_2 + a$ [m] $X_3 = aX_3 + bX_2$ [n] $X_3 = bX_3 + aX_2$
[o] $X_3 = bX_3 + aX_2 + \epsilon$
[p] Aucune des équations proposées. [q] Il manque des données pour répondre à la question.

Question 12 [auto2erbis-2] **(0,5 point)** Nous appliquons le lemme d'Arden sur l'équation associée à X_3. L'équation que nous obtenons pour X_3 est

■ $X_3 = a^*(bX_2 + \epsilon)$ [b] $X_3 = b^*(aX_2 + a)$ [c] $X_3 = b^*(aX_2 + b)$
[d] $X_3 = a^*(bX_2 + b)$ [e] $X_3 = a^+(bX_2 + \epsilon)$ [f] $X_3 = a^+bX_2$
[g] $X_3 = a^+bX_2 + \epsilon$ [h] Aucune des équations proposées.
[i] Il manque des données pour répondre à la question.

Question 13 [auto2erbis-3] **(0,5 point)** Nous utilisons l'équation trouvée pour X_3 à la question précédente dans la définition de X_2, puis simplifions. L'équation que nous obtenons pour X_2 est

■ $X_2 = a^+bX_2 + a^+ + bX_1$ [b] $X_2 = a^*bX_2 + a^+ + bX_1$
[c] $X_2 = a^+baX_2 + a^+ + bX_1$ [d] $X_2 = a^+baX_2 + a^+ + aX_1$
[e] $X_2 = a^+baX_2 + a^* + aX_1$ [f] $X_2 = a^+bX_2 + a^* + aX_1$
[g] Aucune des équations proposées. [h] Il manque des données pour répondre à la question.

Question 14 [auto2erbis-4] **(0,5 point)** Nous utilisons l'équation trouvée pour X_2 à la question précédente et souhaitons lui appliquer le lemme d'Arden.

■ $X_2 = (a^+b)^*(a^+ + bX_1)$ [b] $X_2 = (a^*b)^*(a^+ + bX_1)$ [c] $X_2 = (a^*b)^+(a^+ + bX_1)$
[d] $X_2 = (a^*b)^+(a^+bX_1)$ [e] $X_2 = (a^+b)^+(a^+bX_2)$ [f] $X_2 = (a^+b)^+(a^+bX_3)$
[g] Le lemme d'Arden ne peut pas être appliqué.
[h] Il manque des données pour répondre à la question.

Question 15 [auto2erbis-5] **(0,75 point)** Nous utilisons l'équation trouvée pour X_2 à la question précédente dans l'équation associée à X_1. Nous obtenons :

■ $X_1 = (a(a^+b)^*b + b)X_1 + a(a^+b)^*a^+$ [b] $X_1 = (b(a^+b)^*b + a)X_1 + a(a^+b)^*a^+$
[c] $X_1 = (b(a^+b)^*b + a)X_1 + a(a^*b)^*a^*$ [d] $X_1 = (a(a^+b)^*b + a)X_2 + a(a^*b)^*a^*$
[e] $X_1 = (a(a^+b)^*b + a)X_2 + a(a^+b)^*a^*$ [f] Aucune des équations proposées.
[g] Il manque des données pour répondre à la question.

Partie 4 : Expression régulière vers automate (1 point)

Question 16 [er2auto-1] **(1 point)** Nous considérons la traduction compositionnelle d'expressions régulières vers automates, c'est-à-dire la méthode définie sur la structure de l'expression régulière et qui traduit une expression régulière par composition d'automates pour les sous-expressions régulières. Considérons l'expression régulière $a \cdot a^* \cdot b$. L'ϵ-AEFND résultant de la traduction compositionnelle d'expressions régulières vers automates est celui représenté dans

■ la Figure 4-iv-. b la Figure 4-i-. c la Figure 4-ii-. d la Figure 4-iii-. e aucune des figures proposées.

Partie 5 : Lemme de l'itération et constante d'itération (5 points)

Question 17 [`iteration-cste1`] **(0,5 point)** Considérons le langage dénoté par l'expression régulière

$$a + a \cdot b + a \cdot b \cdot b \cdot b + a^* \cdot b^+.$$

La constante d'itération minimale de ce langage est :

a 0 b 1 ■ 2 d 3 e 4 f 5 g 6

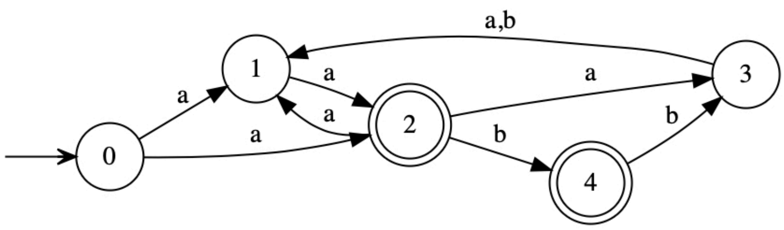
Question 18 [`iteration-cste2`] **(1,5 points)** Démontrer le résultat sur la constante d'itération minimale trouvée à la question précédente.

Question 19 [`iteration-preuve`] **(3 points)** Soit $\Sigma = \{a, b\}$. Démontrer que le langage $\{w \cdot c \cdot w_{\text{pref}} \mid w \in \Sigma^*, w_{\text{pref}} \in \Sigma^*, w_{\text{pref}} \preceq w\}$, sur l'alphabet $\Sigma \cup \{c\}$, est non régulier. Pour rappel, le symbole $\preceq$ fait référence à la relation de préfixe entre mots ; voir le rappel des notations.

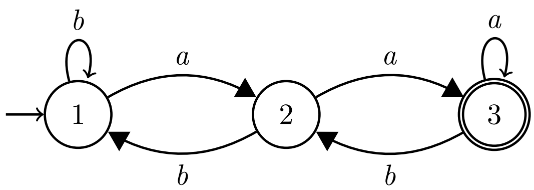
Partie 6 : Algorithmes et transformation d'automates (3 points)

Question 20 [`algo1`] **(1,5 points)** Soit $A = (Q, \Sigma, q_{\text{init}}, \delta, F)$ un AEFD qui reconnaît un langage L. Donner un automate qui reconnaît la fermeture par extension de L.

Question 21 [`algo2`] **(1.5 points)** Soit $A = (Q, \Sigma, q_{\text{init}}, \delta, F)$ un AEFD qui reconnaît un langage L. En ré-utilisant les algorithmes du cours et l'automate trouvé à la question précédente, donner un algorithme qui détermine si le langage reconnu par A est extension-clos. Voir le paragraphe sur les rappels et notations pour la définition de extension-clos.

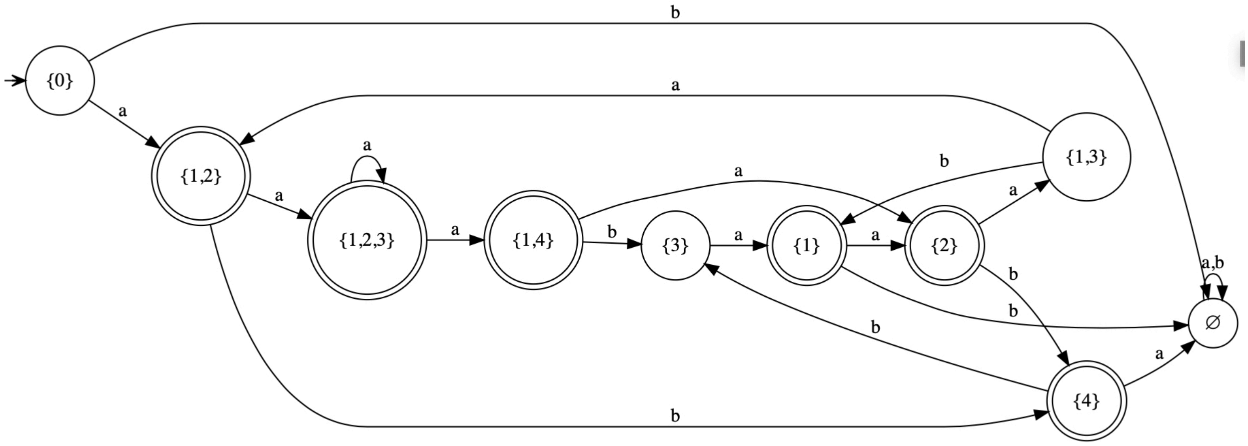


(-i-) Un automate sur lequel on applique l'algorithme de déterminisation (Partie 2).

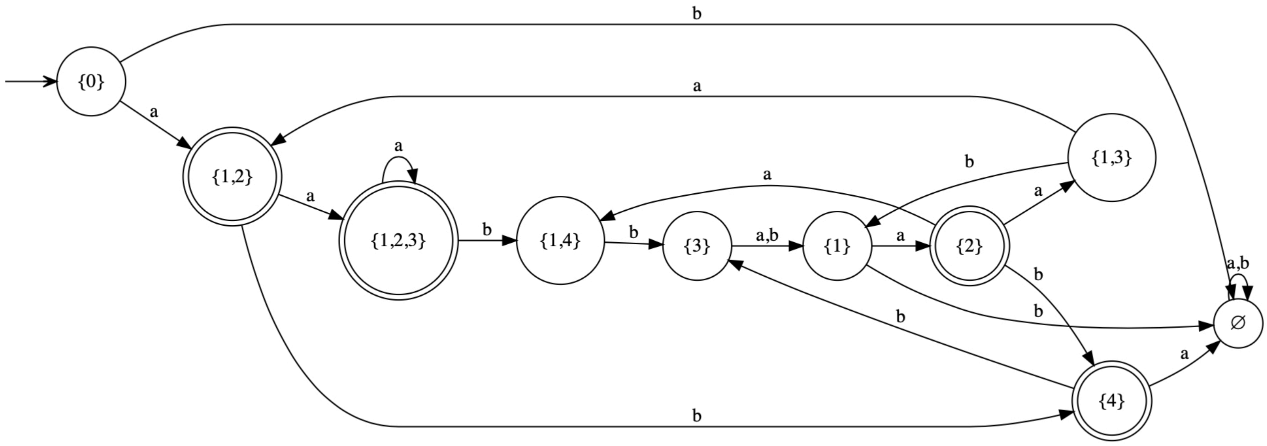


(-ii-) Un automate pour lequel on cherche une expression régulière équivalente (Parties 3 et 3bis).

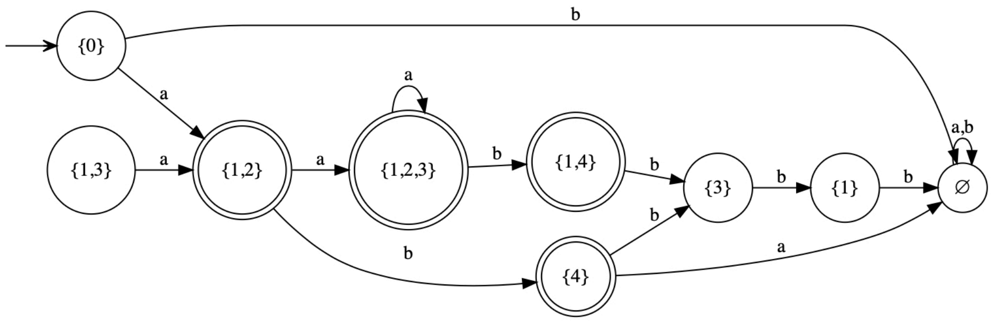
FIGURE 2 – Des automates à utiliser pour les exercices. L'état initial est indiqué par une flèche entrante, sans état source. Les états accepteurs/finaux sont indiqués par des doubles cercles.


{0}
{1,2}
{1,2,3}
{1,4}
{3}
{1}
{2}
{1,3}
{4}
∅

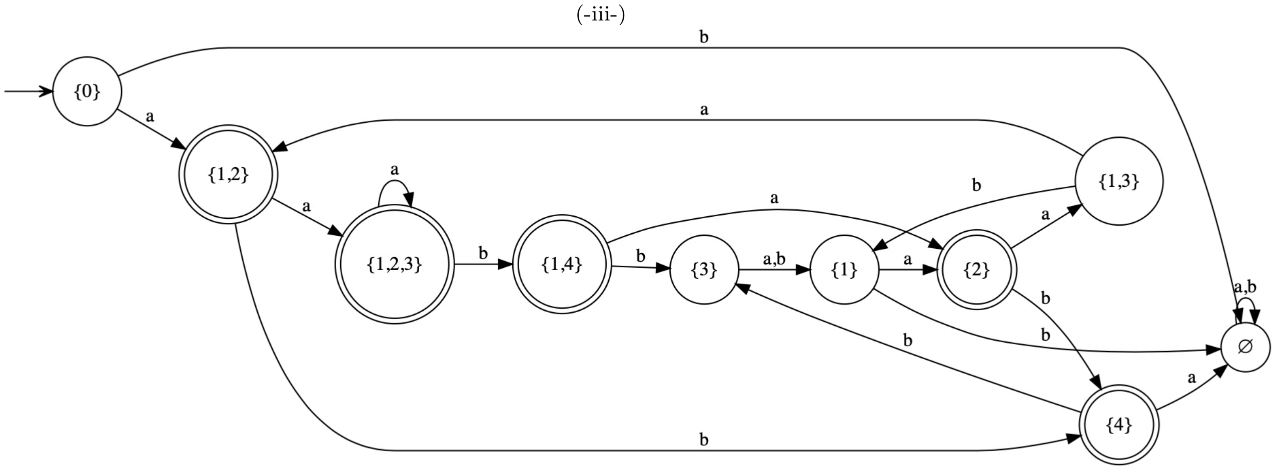
(-i-)

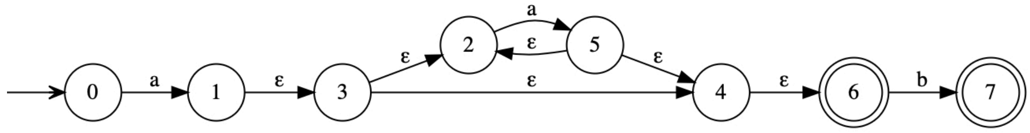

{0}
{1,2}
{1,2,3}
{1,4}
{3}
{1}
{2}
{1,3}
{4}
∅

(-ii-)

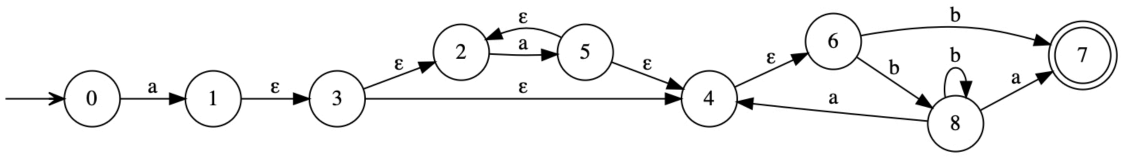

{0}
{1,3}
{1,2}
{1,2,3}
{1,4}
{3}
{1}
{4}
∅

(-iii-)

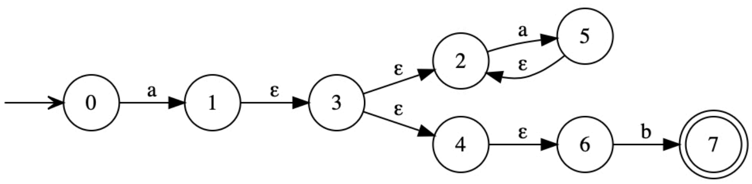

{0}
{1,2}
{1,2,3}
{1,4}
{3}
{1}
{2}
{1,3}
{4}
∅



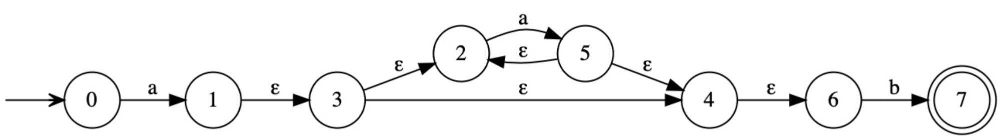
(-i-)



(-ii-)



(-iii-)



(-iv-)

Figure 4 – Des automates résultant possiblement de la traduction compositionnelle d'expressions régulières vers ϵ-AEFND, pour l'expression $a \cdot a^* \cdot b$.

Examen final du 9/01/2020
Licence Sciences et Technologies, 2ème année

INF 302 : Langages et Automates
Année académique 2019/2020

Feuille(s) de réponses

0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9

Codez votre numéro d'anonymat ci-contre et recopiez le manuellement dans la boite.

Numéro d'anonymat :
....

Indiquez la salle d'examen et numéro de place ci-dessous.

Salle d'examen :
....

Numéro de place :
....

Question 1 : ■ *Réservé enseignant*

....

....

....

....

....

....

Question 2 : ■ ■ ■ d e f g h i j

Question 3 : ■ ■ ■ ■ ■ ■ g h i j k l m n o

Question 4 : ■ ■ ■ ■ e f g

Question 5 : a ■ c

Question 6 : ■ ■ ■ ■ ■ f g h i j k l

Question 7 : a b ■

Question 8 : ■ b c d ■ f g h

Question 9 : [redacted] b c d e

Question 10 : [redacted] i j k l m n o p q

Question 11 : [redacted] d e f g h i j k l m n o p q

Question 12 : [redacted] b c d e f g h i

Question 13 : [redacted] b c d e f g h

Question 14 : [redacted] b c d e f g h

Question 15 : [redacted] b c d e f g

Question 16 : [redacted] b c d e

Question 17 : a b [redacted] d e f g

Question 18 : preuve cim f i b [redacted] *Réservé enseignant*

...

Démontrons que la constante d'itération minimale est 2.

- Observons que 1 n'est pas une constante d'itération. En effet, le mot a appartient au langage et est de longueur 1, mais ne peut pas être itéré. (Décomposer a en trois facteurs x, y et z tels que $a = x \cdot y \cdot z$ avec $y \neq \epsilon$ force à prendre $y = a$, mais $x \cdot y^0 \cdot z = \epsilon$ n'appartient pas au langage.)
- Soit w un mot de longueur supérieure ou égale à 2. Distinguons deux cas.
  - si w commence par a, alors w s'écrit sous la forme $w = a^{n_1} \cdot b^{n_2}$ avec $n_1 \geq 1$ et $n_2 \geq 1$. Nous pouvons prendre la décomposition de w en $w = x \cdot y \cdot z$ avec $x = \epsilon$, $y = a$ et $z = a^{n_1 - 1}b^{n_2}$. Cette décomposition vérifie les conditions du lemme de l'itération ($|x \ \cdot y| \leq 2$, $y \neq \epsilon$ et pour tout k, $x \cdot y^k \cdot z$ appartient au langage.
  - si w commence par b, alors w s'écrit sous la forme $w = b^{n_2}$ avec $n_2 \geq 2$. Nous pouvons prendre la décomposition de w en $w = x \cdot y \cdot z$ avec $x = \epsilon$, $y = b$ et $z = b^{n_2 - 1}$. De manière similaire, cette décomposition vérifie les conditions du lemme de l'itération.

Question 19 : preuve non régularité f i ab b *Réservé enseignant*

...

Supposons que le langage soit régulier. Il satisfait donc le lemme de l'itération. Soit N la constante fournie par le lemme de l'itération. Considérons le mot $w = (b \cdot a^N) \cdot c \cdot (b \cdot a^N)$. Le mot w appartient au langage et $|w| \geq N$. Soit x, y, z les trois facteurs donnés par la décomposition de w selon le lemme. Nous distinguons deux cas.

- **Si $x = \epsilon$. Alors y s'écrit $y = a \cdot b^i$, $0 \leq i \leq N-1$. Considérons le mot $w' = x \cdot y^2 \cdot z = b \cdot a^i \cdot b \cdot a^N \cdot c \cdot b \cdot a^N$. Il n'existe aucune décomposition de w' avec deux mots w^1 et w_p^1 tels que $w' = w^1 \cdot c \cdot w_p^1$ et w_p^1 soit un préfixe de w^1.**
- **Si $x \neq \epsilon$. Comme $|x \cdot y| \leq N$ et $y \neq \epsilon$, x et y s'écrivent sous la forme $x = b \cdot a^j$ et $y = a^i$ avec $j \geq 0$ et $i > 0$. Considérons le mot $w' = x \cdot y^0 \cdot z = b \cdot a^{N-i} \cdot c \cdot b \cdot a^N$. Il n'existe aucune décomposition de w' avec deux mots w^1 et w_p^1 tels que $w' = w^1 \cdot c \cdot w_p^1$ et w_p^1 soit un préfixe de w^1.**

Dans les deux cas, nous obtenons une contradiction.

Question 20 :

déf automate fermeture extension f pf pj [redacted] *Réservé enseignant*

...

$(Q, \Sigma, q_{\text{init}}, \delta \cup (F \times \Sigma \times F), F)$

Question 21 : algo langage reconnu extension clos f pf pj *Réservé enseignant*

...

- Calculer l'automate A_{ext} qui est la fermeture par extension de A en utilisant le résultat de la question précédente.
- Vérifier que A et A_{ext} reconnaissent le même langage par exemple en construisant l'automate union et en testant la distinguabilité des anciens états initiaux des automates A_{ext} et A.