IUT de Paris – Dép. Informatique – Théorie des Langages – DST du 6 juin 2017

NOM : RODRIGUES.............................

Prénom : Ruben.......................... Groupe : 107........

Note :

Durée : 2 heures – Sans documents, sans calculatrice, sans portable.

Pour l'exercice 1 et la question 0. de l'exercice 2, vous répondrez directement sur le sujet. La suite de l'exercice 2 sera traitée sur une copie séparée.

Exercice 1 *(∼ 5 points)*

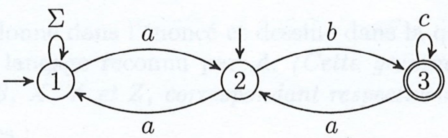
Pour les questions à choix multiples, il peut y avoir une ou plusieurs bonne(s) réponse(s). Toute réponse juste rapportera les points correspondants, et toute réponse fausse fera perdre la moitié de ces points.

Soit l'alphabet $\Sigma = \{a, b, c\}$.

1. On définit le langage $\Lambda = a(a \mid cb)^*bc^*$. Parmi les mots ci-dessous, le(s)quel(s) appartien(nen)t au langage Λ ?

   ☒ $aabc^3$ ☐ cab ☒ ab ☒ a^4cbacb^2c ☐ ε ☐ $abcb$ ☒ $acbc$

2. On définit l'automate $\mathcal{B}$ par son graphe :



   Parmi les mots ci-dessous, le(s)quel(s) appartien(nen)t au langage reconnu par l'automate $\mathcal{B}$?

   ☒ $aabc^3$ ☒ cab ☒ ab ☐ a^4cbacb^2c ☐ ε ☐ $abcb$ ☐ $acbc$

3. On définit la grammaire

$$\Gamma = (\Sigma, NT = \{S, X, Y\}, S, \{S \to aX \mid Yb \mid \varepsilon, X \to aX \mid cbX \mid bY, Y \to Yc \mid c \mid \varepsilon\}).$$

   Parmi les mots ci-dessous, le(s)quel(s) appartien(nen)t au langage engendré par la grammaire G ?

   ☐ cab ☒ ab ☒ a^4cbacb^2c ☒ ε

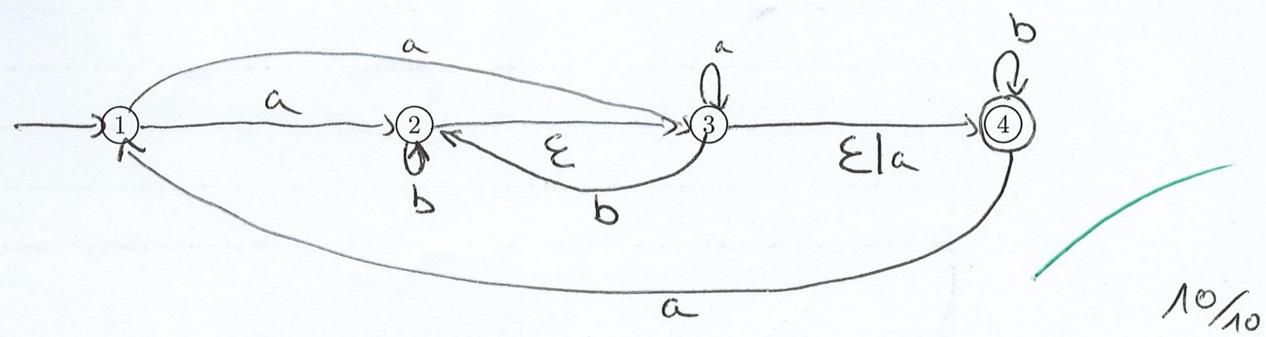
   Pour le mot $aabc^3$, donner une dérivation qui le produit.

Exercice 2 *(∼ 15 points)*

Soit $\Sigma = \{a, b\}$. On définit l'automate fini $\mathcal{A}$ par sa table de transition donnée ci-dessous :

	ε	a	b
- 1		2,3	
2	3		2
3	4	3,4	2
4 +		1	4

0. Dessiner l'automate fini $\mathcal{A}$ sur le modèle ci-dessous :



10/10

Les trois parties de l'exercice sont indépendantes, et peuvent être traitées dans l'ordre qui vous convient. Elles sont à traiter sur une copie séparée.

Partie I :

1. L'automate fini $\mathcal{A}$ est-il déterministe ? complet ? Justifiez vos réponses.
2. En appliquant l'algorithme de "subset construction", donner la table de transition d'un automate fini déterministe complet sans ε-transitions équivalent à $\mathcal{A}$. *(On devrait trouver 5 états.)*
3. En appliquant l'algorithme de Nérode, donner la table de transition et dessiner l'automate fini déterministe complet minimal équivalent à $\mathcal{A}$.
4. En déduire une expression régulière du langage L reconnu par $\mathcal{A}$.
5. Dessiner le graphe d'un automate fini déterministe complet reconnaissant le langage $L' = b\Sigma^* \mid \varepsilon$ des mots ne commençant pas par a.

Partie II : Soit la grammaire

$$G = (\Sigma = \{a, b\}, NT = \{S, X, Y\}, S, P = \{S \xrightarrow{R_1 \mid R_2} aY \mid Xb, X \xrightarrow{R_3} aYa, Y \xrightarrow{R_4 \mid R_5 \mid R_6} bY \mid aY \mid \varepsilon\}).$$

1. Cette grammaire est-elle algébrique ? linéaire ? régulière ? Justifiez vos réponses.
2. Peut-on utiliser une chaîne de production pour la grammaire G ? Justifiez. Si oui, dessiner la chaîne de production correspondante.
3. Déterminer le langage engendré par la grammaire G.
4. Ce langage est-il régulier ? Si non, justifiez. Si oui, donner une expression régulière de ce langage.

Partie III :

1. A partir de l'automate fini $\mathcal{A}$ donné dans l'énoncé et dessiné dans la question 0., déterminer une grammaire régulière engendrant le langage reconnu par $\mathcal{A}$. *(Cette grammaire comportera* 4 *symboles non terminaux, que l'on appellera* S, X, Y *et* Z, *correspondant respectivement aux états* 1, 2, 3 *et* 4.*)*
2. Démontrer les égalités suivantes :
   (a) $(ab^* \mid a) = ab^*$
   (b) $(bb^* \mid a)^* = \Sigma^*$
   On pourra admettre les égalités suivantes :
   (c) $b^*\Sigma^*(\varepsilon \mid a) = \Sigma^*$
   (d) $(a\Sigma^*a)^*a\Sigma^* = a\Sigma^*$
3. (a) En appliquant le théorème d'Arden, exprimer X en fonction de Y. En déduire S en fonction de Y. *(On pourra simplifier l'expression en utilisant les égalités de la question précédente.)*
   (b) En appliquant le théorème d'Arden, exprimer Y en fonction de Z. En déduire S en fonction de Z. *(On pourra simplifier l'expression en utilisant les égalités de la question précédente.)*
   (c) Appliquer une nouvelle fois le théorème d'Arden pour exprimer Z en fonction de S.
   (d) Appliquer une dernière fois le théorème d'Arden et en déduire le langage engendré par la grammaire déterminée dans la question 1. *(On pourra simplifier l'expression en utilisant les égalités de la question précédente.)*

UNIVERSITÉ PARIS DESCARTES

IUT

DÉPARTEMENT INFORMATIQUE

Écrire très lisiblement

NOM : RODRIGUES (en capitales)

Prénom : Ruben

DISCIPLINE : Langage

Date de l'épreuve : 06/06/2017

Année : 1A Groupe : 107

NOTE DE 0 À 20	APPRÉCIATIONS
	4.257 (0.5+5.5+2+4) = 16.5

Ne rien écrire dans cette marge

Partie I :

1) Un automate ~~et~~ fini est déterministe s'il admet un ~~unique~~ au plus état de départ et au maximum 1 état dans chaque case de sa table de transition.

Ici à la première ligne on a :

	ε	a	b
→1		2,3	

L'automate A n'est donc pas déterministe.

Un automate fini est complet s'il admet au moins un état pour chaque transition possible; et au moins un état de départ

D'après sa table de transition, ce n'est pas le cas pour l'automate A, celui-ci n'est donc pas complet

2) On cherche a déterminer un automate déterministe complet sans ε-transitions équivalent à A, on utilise l'algorithme di "Subset Construction":

1/18

On détermine l'ensemble de départ D :

$\{1\} \cdot \varepsilon = \emptyset$

$D = \{1\}$

	a	b
$-\ \{1\}_1$	$\{2,3,4\}_2$	$\emptyset_3$
$\{2,3,4\}^+_2$	$\{1,3,4\}_4$	$\{2,3,4\}_2$
$\emptyset_3$	$\emptyset_3$	$\emptyset_3$
$\{1,3,4\}^+_4$	$\{1,2,3,4\}_5$	$\{2,3,4\}_2$
$\{1,2,3,4\}^+_5$	$\{1,2,3,4\}_5$	$\{2,3,4\}_2$

Justifications :

$\{1\} \cdot a = \{2,3\}$

$\{2,3\} \cdot \varepsilon = \{3,4\}$

$\{3,4\} \cdot \varepsilon = \{4\}$

$\{4\} \cdot \varepsilon = \emptyset$

$\{1\} \cdot b = \emptyset$

$\{2,3,4\} \cdot a = \{1,3,4\}$

$\{1,3,4\} \cdot \varepsilon = \{4\}$

$\{4\} \cdot \varepsilon = \emptyset$

$\{2,3,4\} \cdot b = \{2,4\}$

$\{2,4\} \cdot \varepsilon = \{3\}$

$\{3\} \cdot \varepsilon = \{4\}$

$\{4\} \cdot \varepsilon = \emptyset$

On obtient :

	a	b
−1	2	3
2+	4	2
3	3	3
4+	5	2
5+	5	2

3) On cherche à déterminer l'automate fini déterministe complet équivalent à A, pour cela on utilise l'algorithme de Nerode :

On détermine les ~~états~~ ensembles A et $\bar{A}$ d'états acceptants et non acceptants :

$A = \{2, 4, 5\}$

$\bar{A} = \{1, 3\}$

Et L la liste des couples d'états acceptants et non acceptants :

$L = \{(2,4), (2,5), (4,5), (1,3)\}$

2

10

On utilise l'algorithme de Nerode :

$L_0 = L$

	L_0	a	b
L_1	2,4	{4,5}	{2}
	2,5	{4,5}	{2}
	4,5	{5}	{2}
	~~1,3~~	~~{2,3}~~	~~{3}~~

$(2,3) \notin L$

	L_1	a	b
L_2	2,4	{4,5}	{2}
	2,5	{4,5}	{2}
	4,5	{5}	{2}

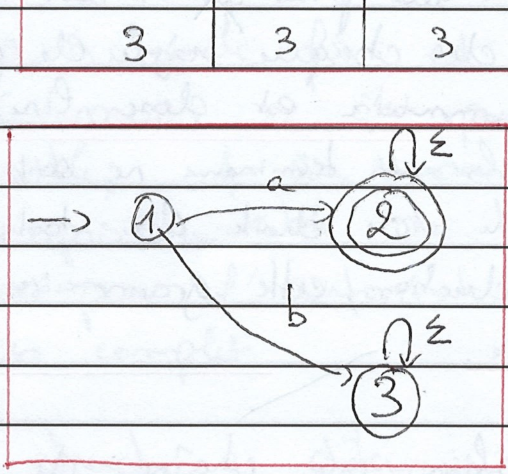
	L_2	a	b
L_3	2,4	{4,5}	{2}
	2,5	{4,5}	{2}
	4,5	{5}	{2}

$L_3 = L_2$ On arrête l'algorithme la liste étant stable

On peut donc regrouper les états 2, 4, 5 en un seul état ; on obtient la table de transition suivante :

	a	b
− 1	2	3
2 +	2	2
3	3	3

A'' :



4) On en déduit l'expression régulière du langage suivant :

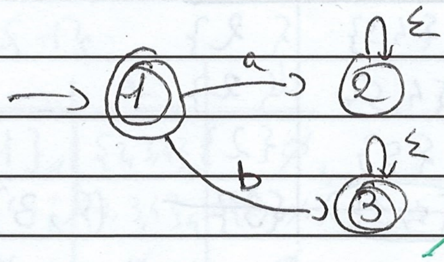
$L(A'') = L(A) = a\Sigma^*$

3/10

5) $L' = b\Sigma^* | \varepsilon$

On inverse les états acceptants et non acceptants de A'' :
$\overline{A''}$:

pourquoi ?



On détermine $L(\overline{A''}) = b\Sigma^* | \varepsilon$

$\boxed{L(\overline{A''}) = L'}$

L'automate $\overline{A''}$ reconnait donc le langage L'

Partie II :

$G = (\Sigma = \{a, b\}, NT = \{S, X, Y\}, S,$
$P = \{ S \rightarrow aY | Xb,$
$X \rightarrow aYa,$
$Y \rightarrow bY | aY | \varepsilon \})$

1) Il n'y a qu'un seul et unique symbole NT à gauche de chaque règle de production, cette grammaire est donc algébrique
Il ~~n~~'y a au plus ~~qu~~'un seul symbole NT à droite de chaque règle de production, cette grammaire est donc linéaire
Les symboles non terminaux ne sont pas tout à droite du membre de droite dans toutes les ~~exps~~ règles de production, cette grammaire n'est donc pas régulière.

2) Pour utiliser une chaîne de production il faut que la grammaire soit linéaire ce qui est le cas ici, on peut donc utiliser une chaîne de production

UNIVERSITÉ PARIS DESCARTES
IUT
DÉPARTEMENT INFORMATIQUE

DISCIPLINE : Langage

Date de l'épreuve : 06/06/2017

Année : 1A **Groupe :** 107

NOM : RODRIGUES (en capitales)

Prénom : Ruben

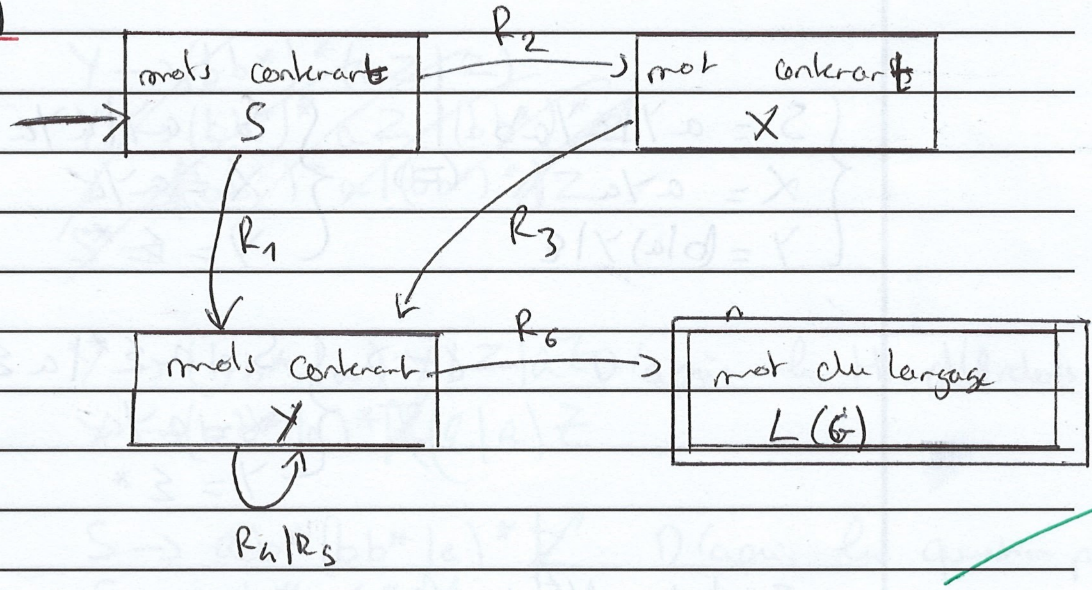
Écrire très lisiblement

NOTE DE 0 À 20	APPRÉCIATIONS

Ne rien écrire dans cette marge

Partie II (suite) :

2)



3) On détermine le langage engendré par cette grammaire :
C'est le langage des mots commençant par a (...)

4) Oui ce langage est ~~régulier~~. à justifier

5/10

On pose les règles de production sous forme d'un système que l'on résout pour obtenir l'expression régulière du langage :

$$\begin{cases} S = aY \mid Xb \\ X = aYa \\ Y = bY \mid aY \mid \varepsilon \end{cases} \iff \begin{cases} S = aY \mid aYab \\ X = aYa \\ Y = bY \mid aY \mid \varepsilon = (b|a)Y \mid \varepsilon \end{cases}$$

$$\iff \cancel{\begin{cases} S = a(bY|aY|\varepsilon) \mid aYab \\ X = aYa \\ Y = bY \mid aY \mid \varepsilon \end{cases}}$$

$$\iff \cancel{\begin{cases} S = abY \mid aaY \mid a \mid aYab \\ X = aYa \\ Y = bY \mid aY \mid \varepsilon \end{cases}}$$

or d'après le théorème d'Arden :

$$\boxed{\begin{array}{l} L = xL \mid y \text{ avec } \varepsilon \notin x \\ L = x^* y \end{array}}$$

donc $Y = (b|a)^*$
$Y = \Sigma^*$

$$\begin{cases} S = aY \mid aYab \\ X = aYa \\ Y = (b|a)Y \mid \varepsilon \end{cases} \iff \begin{cases} S = aY \mid aYab \\ X = aYa \\ Y = \Sigma^* \end{cases}$$

$$\iff \begin{cases} S = a\Sigma^* \mid a\Sigma^* ab = a\Sigma^*(ab|\varepsilon) \\ X = aYa \\ Y = \Sigma^* \end{cases}$$

S étant l'état de départ, on en déduit l'expression régulière de la grammaire G suivante :

$$\boxed{L(G) = a\Sigma^*(ab|\varepsilon)}$$

Ok, on peut faire plus simple avec les règles de production ...

Partie III :

On détermine la grammaire G_1 d'après l'automate A de la question 0 ; on obtient :

1) $G_1 = (\Sigma = \{a, b\}, NT = \{S, X, Y, Z\}, S,$
$P = \{S \to aX \mid aY,$
$X \to bX \mid Y$
$Y \to bX \mid a\cancel{X} \mid Z \mid aZ$
$Z \to bZ \mid aS \mid \varepsilon\})$

2) $(ab^* \mid a) = ab^*$ revient à dire que :

$ab^* \mid a \subset ab^*$ et $ab^* \subset ab^* \mid a$

Les expressions régulières représentent des ensembles.

a) 1) $ab^* \subset ab^* \mid a$:

Ici c'est évident, car b^* est inclus dans $ab^* \mid a$

a) 2) $ab^* \mid a \subset ab^*$

$a = ab^0$, or $b^0 \subset b^*$ (etoile de Kleen)

donc $ab^0 \subset ab^*$
donc $(ab^* \mid ab^0) \subset ab^*$
donc $ab^* \mid a \subset ab^*$
Ou $ab^* = ab^* \mid a$
donc $ab^* \subset ab^*$

On a donc bien $ab^* \mid a = ab^*$

7/10

b) $(bb^*|a)^* = \Sigma^*$

$(bb^*|a)^* = (a|b)^*$

Ça revient à : $(bb^*|a^*)^* \subset (a|b)^*$

et $(a|b^*)^* \subset (bb^*|a^*)$

b) 1) $(bb^*|a)^* \subset (a|b)^*$

nsa !

$(bb^*|a)^* \subset (a|b)^* \Leftrightarrow (bb^*)^*|a^* \subset (a|b)^*$

or on voit que $(bb^*)^* \subset (a|b)^*$

et on voit que $a^* \subset (a|b)^*$

donc $(bb^*|a)^* \subset (a|b)^*$

2) $(a|b)^* \subset (bb^*|a)^*$

L'inclusion est évidente ici. ??

On a donc $(bb^*|a)^* = \Sigma^*$

3) (a) $X \to bX|Y$

D'après le théorème d'Arden :

$L = xL|y$ avec $\varepsilon \notin x$

alors $L = x^*y$

On a donc ici : $X = b^*Y$ oui

$S \to aX|aY$ on remplace X par sa valeur dans S

$S \to ab^*Y|aY$

$S \to (ab^*|a)Y$ D'après la question précédente $ab^*|a = ab^*$

$S \to ab^*Y$

(b) $Y \to bX|aZ|Z|aZ$ On remplace X par sa valeur

$Y \to bb^*Y|aZ|Z|aZ$ ~~D'après le théorème d'Arden~~

Y ~~$\to (bb^*)^*(aZ|Z)$~~

UNIVERSITÉ
PARIS DESCARTES
IUT
DÉPARTEMENT INFORMATIQUE

Écrire très lisiblement

NOM : RODRIGUES (en capitales)

Prénom : Ruben

DISCIPLINE : Langage

Date de l'épreuve : 06/06/2017

Année : 1A Groupe : 102

NOTE DE 0 À 20	APPRÉCIATIONS

Ne rien écrire dans cette marge

Partie III (suite) :

3) ~~$Y \to (bb^*)^*(aZ|Z)$~~
~~$Y \to (bb^*)^* aZ | (bb^*)^* Z$~~
~~$Y \to (bb^*)^* a | (bb^*)^* Z$~~
~~$S \to$~~

$Y \to (bb^*|a)Y|Z|aZ$ D'après le th d'Arden
$Y \to (bb^*|a)^*\cancel{Z}(\varepsilon|a)Z$

$S \to ab^*(bb^*|a)^*\cancel{Z}$ D'après la question précédente
$S \to ab^* \Sigma^* \cancel{Z}$

c) $Z \to bZ|aS|\varepsilon$ D'après le théorème d'Arden
$Z \to b^*(aS|\varepsilon)$
$Z \to b^*aS|b^*$

Oui

d) $S \to ab^*\Sigma^*(b^*aS|b^*)$
$S \to ab^*\Sigma^*b^*aS|ab^*\Sigma^*b^*b^*$ D'après le th d'Arden
$S \to (ab^*\Sigma^*b^*a)^* ab^*\Sigma^*b^*b^*$

méthode OK

modulo l'erreur ci-dessus

9/18