Examen Langages et Systèmes Formels

Durée: 1h45.

Cet examen comporte 7 exercices indépendants sur 2 pages.
Les exercises peuvent être traités dans n'importe quel ordre.
Toute réponse doit être justifiée.
Le barème est donné à titre indicatif: il est susceptible d'être modifié.
Le total est sur 20 points.
Poly, notes de cours et TDs autorisés. Prêts entre voisins interdits.

Exercice 1 (2 points) Montrez que deux mots u et v commutent ($uv = vu$) si et seulement s'il existe un mot w et des entiers i et j tels que $u = w^i$ et $v = w^j$.

Solution.

- $\Leftarrow$: $uv = w^i w^j = w^j w^i = vu$.
- $\Rightarrow$: Par récurrence sur la longueur de uv. Si $|uv| = 0$, alors $u = v = \varepsilon = w^0$. Si $u = \varepsilon$ ou $v = \varepsilon$, c'est immédiat. Si $|u| = |v|$, alors $u = v$ et c'est également fini. Reste le cas où $|u| \neq |v|$. Supposons $|u| < |v|$ (le cas $|u| > |v|$ est similaire). Alors, il existe un mot x tel que $v = ux$ et $ux = xu$. Comme $|x| < |v|$, par hypothèse de récurrence, $u = w^i$ et $x = w^j$. Donc, $v = ux = w^i w^j = w^{i+j}$.$\blacksquare$

Exercice 2 (3 points) Soit G la grammaire suivante:

$$\begin{array}{rcl} S & \to & aA|bBB|abAS|BC \\ A & \to & aB|bA|a \\ B & \to & aB|bB \\ C & \to & aA|bS|a \end{array}$$

où S est le non-terminal de départ, A, B, C sont des non-terminaux, et a et b des terminaux. Donner une grammaire réduite équivalente en appliquant la transformation vue en cours. On prendra soin d'expliquer chaque étape et de justifier toute affirmation.

Solution.

1. On calcule l'ensemble $\mathcal{P}$ des non-terminaux productifs.
   - A et C sont productifs car $A \to a$ et $C \to a$.
   - Donc S est productifs car $S \to aA$.
   - Aucun autre non-terminal n'est productif.
2. On élimine les règles contenant des non-terminaux improductifs, en l'occurrence B:

$$\begin{array}{rcl} S & \to & aA|abAS \\ A & \to & bA|a \\ C & \to & aA|bS|a \end{array}$$

3. On calcule les non-terminaux accessibles.
   - S est accessible par définition.
   - Donc A est accessible puisque $S \to aA$.
   - Aucun autre non-terminal n'est accessible.
4. On élimine les règles contenant des non-terminaux inaccessibles, ici C:

$$\begin{array}{lcl} S & \to & aA|abAS \\ A & \to & bA|a \end{array}$$

$\blacksquare$

Exercice 3 (3 points) Soit l'alphabet $\Sigma = \{a, b\}$. Soit $L = \{u^2 \mid u \in \Sigma^*\}$ le langage des carrés sur Σ, et $M = \Sigma^* - L$ son complémentaire.

(a) Montrez que tout mot de longueur impaire est dans M.

(b) Montrez qu'un mot $x_1 \dots x_{2n}$ de longueur $2n$ est dans M si et seulement s'il existe i tel que $x_i \neq x_{n+i}$.

(c) Soit G la grammaire:
$S \to A|B|AB|BA$
$A \to a|aAa|aAb|bAa|bAb$
$B \to b|aBa|aBb|bBa|bBb$

Décrire en français le langage $L_G(A)$ des mots dérivables à partir de A, et le langage $L_G(B)$ des mots dérivables à partir de B.

(d) Montrez que tout mot de longueur impaire est dans $L(G) = L_G(S)$.

(e) Montrez que tout mot de M de longueur paire est dans $L(G)$.

(f) Montrez que tout mot de $L(G)$ de longueur paire est dans M.

(g) En déduire que M est algébrique.

Solution.

(a) Un carré est de longueur paire.

(b) $x = x_1 \dots x_{2n}$ est un carré si et seulement si, pour tout $i \in \{1, \dots, n\}$, $x_i = x_{n+i}$.

(c) $L_G(A)$ est l'ensemble des mots de longueur impaire dont le caractère du milieu est a. $L_G(B)$ est l'ensemble des mots de longueur impaire dont le caractère du milieu est b.

(d) Si u est un mot de longueur impaire et centre a, alors $S \to A \to^* u$. Si u est un mot de longueur impaire et centre b, alors $S \to B \to^* u$.

(e) Soit $x = x_1 \dots x_{2n}$ un mot de M de longueur paire. Alors il existe i tel que $x_i \neq x_{n+i}$. Par ailleurs, $y = x_1 \dots x_{2i-1}$ est un mot de longueur impaire et centre x_i, et $z = x_{2i} \dots x_{2n}$ est un mot de longueur impaire et centre x_{n+i}. Sans perte de généralité, on peut supposer que $x_i = a$ et $x_{n+i} = b$. Alors, $A \to^* y$ et $B \to^* z$. Donc, $S \to AB \to^* yz$ et $x \in L(G)$.

(f) Soit x un mot de $L(G)$ de longueur paire. Comme x est de longueur paire, on doit avoir $S \to AB \to^* x$ ou $S \to BA \to^* x$. Dans le premier cas, on a x de la forme $uavu'bv'$ d'après (c). Dans le second cas, on x de la forme $ubvu'av'$ d'après (c). Donc on a $x \in M$ d'après (a).

(g) Etant donné un language Q, soit Q_i l'ensemble des mots de Q de longueur impaire, et Q_p l'ensemble des mots de P de longueur paire. Nous avons $Q = Q_i \uplus Q_p$.

D'après (a), $M = I \uplus M_p$ où I est l'ensemble des mots de longueur impaire. D'après (d), $I \subseteq M'$ où $M' = L(G)$. D'après (e), $M_p \subseteq M'_p$. Donc, $M \subseteq M'$. D'après (f), $M'_p \subseteq M_p$. Donc, $M'_p = M_p$ et $M'_i = I$. Il s'ensuit que $M = L(G)$ et donc que M est algébrique car G est algébrique. ■

Exercice 4 (3 points) Minimiser l'automate suivant en appliquant la transformation vue en cours. On prendra soin d'expliquer chaque étape et de justifier toute affirmation. On présentera l'automate résultant dans une table avec une ligne pour chaque lettre de l'alphabet et une colonne pour chaque état. On n'oubliera pas d'indiquer les états initiaux et finaux.

	A	B	C	D	E	F	G	H
0	B	G	A	C	H	C	G	G
1	F	C	C	G	F	G	E	C

Etat initial: A. Etat final: H.

Solution. On remarque que l'automate est déterministe et complet. On calcule les états accessibles. On voit que l'état D est inaccessible. On peut donc l'éliminer.

L'ensemble des états est donc $Q = \{A, B, C, E, F, G, H\}$. $Q/\simeq_0 = \{a, b\}$ où $a = \{H\}$ est l'ensemble des états finaux et $b = \{A, B, C, E, F, G\}$ est l'ensemble des états non finaux. a ne peut pas être raffiné davantage car c'est un singleton. Pour b, on remplace dans la table de transition les états par leurs classes d'équivalence modulo $\simeq_0$:

	A	B	C	E	F	G
0	b	b	b	a	b	b
1	b	b	b	b	b	b

On a donc $Q/\simeq_1 = \{a, c, d\}$ avec $c = \{E\}$ et $d = \{A, B, C, F, G\}$. c ne peut pas être raffiné davantage. Pour d, on remplace dans la table de transition les états par leurs classes d'équivalence modulo $\simeq_1$:

	A	B	C	F	G
0	d	d	d	d	d
1	d	d	d	d	c

On a donc $Q/\simeq_2 = \{a, c, e, f\}$ avec $e = \{G\}$ et $f = \{A, B, C, F\}$. e ne peut pas être raffiné davantage. Pour f, on remplace dans la table de transition les états par leurs classes d'équivalence modulo $\simeq_2$:

	A	B	C	F
0	f	e	f	f
1	f	f	f	e

On a donc $Q/\simeq_3 = \{a, c, e, g, h, i\}$ avec $g = \{B\}$, $h = \{A, C\}$ et $i = \{F\}$. g et i ne peuvent pas être raffinés davantage. Pour h, on remplace dans la table de transition les états par leurs classes d'équivalence modulo $\simeq_3$:

	A	C
0	g	h
1	i	h

On a donc $Q/\simeq_4 = \{a, c, e, g, i, j, k\}$ avec $j = \{A\}$ et $k = \{C\}$. j et k ne peuvent pas être raffinés davantage. On a donc atteint le point fixe. L'automate sans l'état D est donc minimal parmi tous les automates déterministes complets accessibles représentant le même langage. ■

Exercice 5 (3 points) Eliminer les ε-transitions de l'automate suivant. On présentera l'automate résultant dans une table avec une ligne pour chaque lettre de l'alphabet et une colonne pour chaque état. On n'oubliera pas d'indiquer les états initiaux et finaux.

	p	q	r	s	t	u	v	w	x	y	z
0	q		s				w				
1		r					u	x		z	
ε	w	v		t	q	t	u		y		

Etat initial: p. Etat final: z.

Solution.

1. On calcule les ε-réduits de chacun des états:

	p	q	r	s	t	u	v	w	x	y	z
$\xrightarrow{\varepsilon}^*$	pw	$qtuv$	r	$qstuv$	$qtuv$	$qtuv$	$qtuv$	w	xy	y	z

2. On ajoute dans les états initiaux tous les réduits d'un état initial: p, w.

3. Pour chaque transition $\alpha \xrightarrow{k} \beta$ et ε-réduit β' de β, on rajoute une transition $\alpha \xrightarrow{k} \beta'$:

	p	q	r	s	t	u	v	w	x	y	z
0	$qtuv$		$qstuv$				w				
1		r					$qtuv$	xy		z	

■

Exercice 6 (3 points) Soit G la grammaire des instructions (très simplifiées) délivrées par un logiciel de calcul d'itinéraire (C est le non-terminal de départ):

$$C \to I|IC \quad I \to a|ST \quad T \to g|d \quad S \to \varepsilon|p$$

où C =chemin, I =instruction, a =avancer, S =signe, T =tourner, g =gauche, d =droite, p =panneau.

(a) Cette grammaire est-elle $LL(1)$? Pourquoi?

(b) Donner une grammaire équivalente qui vous paraît $LL(1)$. Montrez qu'elle est bien équivalente.

(c) Construire la table de transition en détaillant les calculs de Pre_1 et $Suiv_1$.

(d) La grammaire est-elle $LL(1)$? Pourquoi?

(e) Donnez l'exécution de l'automate à pile sur apg. Qu'en concluez-vous?

(f) Donnez l'exécution de l'automate à pile sur agp. Qu'en concluez-vous?

Solution.

(a) Cette grammaire n'est pas $LL(1)$ car les membres droits des règles de C ont un préfixe commun I.

(b) Nous proposons de remplacer les règles $C \to I|IC$ par:
$C \to IC'$
$C' \to \varepsilon|IC'$
Soit G' la grammaire ainsi obtenue. Nous avons $L_G(C) = L_G(I)^+$, $L_{G'}(C) = L_{G'}(I)L_{G'}(C')$ et $L_{G'}(C') = L_{G'}(I)^*$. Or, $L_G(I) = L_{G'}(I)$. Donc, $L(G) = L(G')$.

(c) Pour chaque règle $X \to r$, on calcule $Pre_1(rSuiv_1(X))$:

- $C \to IC'$. $Pre_1(IC'Suiv_1(C)) = Pre_1(I) = \{a, g, d, p\}$.
- $C' \to \varepsilon$. $Pre_1(\varepsilon Suiv_1(C')) = \{\varepsilon\}$.
- $C' \to IC'$. $Pre_1(IC'Suiv_1(C')) = Pre_1(I) = \{a, g, d, p\}$.
- $I \to a$. $Pre_1(aSuiv_1(I)) = \{a\}$.
- $I \to ST$. $Pre_1(STSuiv_1(I)) = \{g, d, p\}$
- $T \to g$. $Pre_1(gSuiv_1(T)) = \{g\}$.
- $T \to d$. $Pre_1(dSuiv_1(T)) = \{d\}$.
- $S \to \varepsilon$. $Pre_1(\varepsilon Suiv_1(S)) = \{g, d\}$.
- $S \to p$. $Pre_1(pSuiv_1(S)) = \{p\}$.

On obtient ainsi la table:

	a	g	d	p	ε
C	IC'	IC'	IC'	IC'	
C'	IC'	IC'	IC'	IC'	ε
I	a	ST	ST	ST	
T		g	d		
S		ε	ε	p	

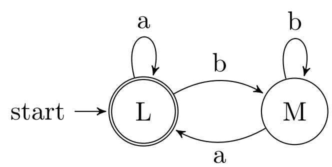
Les cases vides sont des cases d'erreur.

(d) Comme il n'y a qu'une règle par case, la grammaire est $LL(1)$.

(e) $(C, apg) \to (IC', apg) \to (aC', apg) \to (C', pg) \to (IC', pg) \to (STC', pg) \to (pTC', pg) \to (TC', g) \to (gC', g) \to (C', \varepsilon) \to (\varepsilon, \varepsilon)$. Donc, le mot apg appartient à $L(G)$.

(f) $(C, agp) \to (IC', agp) \to (aC', agp) \to (C', gp) \to (IC', gp) \to (STC', gp) \to (TC', gp) \to (gC', gp) \to (C', p) \to (IC', p) \to (STC', p) \to (pTC', p) \to (TC', \varepsilon)$. Nous arrivons dans un état d'erreur. Donc, $agp \notin L(G)$. ■

Exercice 7 (3 points) Exprimez le language reconnu par l'automate suivant par une expression régulière. Justifiez votre réponse.



Solution. Les languages reconnus à partir de L et M vérifie les équations suivantes:
(1) $L = \varepsilon + aL + bM$
(2) $M = bM + aL$

Par le lemme d'Arden, comme $\varepsilon \notin b$, (2) n'a qu'une seule solution: $M = b^*aL$. Ainsi, $L = \varepsilon + aL + b^+aL = b^*aL + \varepsilon$ (3). Par le lemme d'Arden, comme $\varepsilon \notin b^*a$, (3) n'a qu'une seule solution: $L = (b^*a)^*$.

Autre expression possible: Par le lemme d'Arden, comme $\varepsilon \notin a$, (1) n'a qu'une seule solution: $L = a^*(\varepsilon + bM)$. Ainsi, $M = bM + a^+ + a^+bM = a^*bM + a^+$ (4). Par le lemme d'Arden, comme $\varepsilon \notin a^*b$, (4) n'a qu'une seule solution: $M = (a^*b)^*a^+$. Donc, $L = a^* + a^*b(a^*b)^*a^+ = a^* + (a^*b)^+a^+$. ■