ISIMA 1ère année

3 Septembre 1998
Durée : 2 heures
Documents autorisés

AUTOMATES

Première partie :

Soit $V = \{a , b\}$. Soit L le langage sur V contenant les mots commençant par *ab* et suivi par n'importe quel mot de V^* qui ne comporte pas le sous-mot *ab* :
$L = \{ ab, aba, abb, abaa, abba, abbb, abaaa, abbaa, abbbb, \ldots \}$
Soit $\overline{L} = V^* - L = \{ \varepsilon, a, b, aa, ba, bb, aaa, aab, baa, bab, bba, bbb, aaaa, \ldots , abab, \ldots \}$

1°) Construire un AFD, M, acceptant le langage L.
Solution :

2°) Minimiser l'automate obtenu (ou bien, le cas échéant, prouver que l'automate obtenu est minimal).
Solution :
Classes de 0-équivalence : {1, 2, 6}, {3, 4, 5};
Classes de 1-équivalence : {1, 6}, {2}, {3, 5}, {4};
Classes de 2-équivalence : {1}, {6}, {2}, {3}, {5}, {4};
L'AFD obtenu est donc minimal.

3°) Donner l'expression régulière représentant le langage L.
Solution : ab^+a^*

4°) Déduire de M un AFD acceptant $\overline{L}$. Cet AFD est-il minimal ?
Solution :
C'est le même automate, en intervertissant états finaux et non finaux, et il est donc également minimal, puisque les classes de *k*-équivalence sont les mêmes, ce n'est que leur nature qui est inversée.

5°) Donner l'expression régulière représentant le langage $\overline{L}$.

<u>Solution</u> : $\varepsilon + a + b + ab^+a^*b(a+b)^*$

<u>Deuxième partie</u> :

Soit $V = \{a , b\}$. Soit L le langage sur V des mots constitués par la concaténation de deux mots de V^* : $L = \{ ww, w \in V^* \}$

1°) Montrer que $\forall\ W = ww \in L$, il existe une décomposition de $W = uvxyz$, $v \neq \varepsilon$, $y \neq \varepsilon$ telle que $\forall\ n \in \mathbb{N} : uv^nxy^nz \in L$. Que peut-on en déduire ?
<u>Solution</u> :
Il suffit que v et y correspondent à la même partie de w pour que $\forall\ n \in \mathbb{N} : uv^nxy^nz \in L$. On ne peut rien en déduire, car le théorème du gonflement indique que cette propriété nécessaire pour qu'un langage soit hors contexte, mais elle n'est pas suffisante.

2°) Pensez vous que L soit un langage hors contexte ? Pourquoi ?
<u>Solution</u> :
L n'est pas un langage hors contexte, car il n'est pas possible de construire un APND acceptant L. En effet, un APND ne peut empiler les lettres que dans le sens de la lecture, et par conséquent les dépiler dans le sens inverse de la lecture. Or pour reconnaître L, il faudrait pouvoir dépiler également dans le sens de la lecture.

3°) Construire une machine de Turing acceptant le langage $L' = \{>\}L$ (c'est à dire le langage des mots de L précédés par le caractère ">"), basée sur l'algorithme suivant : marquer, par exemple, la première lettre du mot a ou b respectivement par A ou B, puis marquer la dernière lettre du mot a ou b respectivement par α ou β; recommencer cette opération avec la deuxième et l'avant dernière lettre, et ainsi de suite jusqu'à ce que toutes les lettres du mot aient été marquées; il suffit alors de comparer les deux demi-mots de $\{A, B\}^*$ et $\{\alpha, \beta\}^*$.
<u>Solution</u> :

	a	b	A	B	α	β	*	>	#
q_0	(q_1,A,R)	(q_1,B,R)			$(q_{4\alpha},*,L)$	$(q_{4\beta},*,L)$		$(q_0,>,R)$	$(q_f,\#,R)$
q_1	(q_1,a,R)	(q_1,b,R)			(q_2,α,L)	(q_2,β,L)			$(q_2,\#,L)$
q_2	(q_3,α,L)	(q_3,β,L)							
q_3	(q_3,a,L)	(q_3,b,L)	(q_0,A,R)	(q_0,B,R)					
$q_{4\alpha}$			$(q_{4\alpha},A,L)$	$(q_{4\alpha},B,L)$			$(q_{5\alpha},*,R)$	$(q_{5\alpha},>,R)$	
$q_{4\beta}$			$(q_{4\beta},A,L)$	$(q_{4\beta},B,L)$			$(q_{5\beta},*,R)$	$(q_{5\beta},>,R)$	
$q_{5\alpha}$			$(q_6,*,R)$						
$q_{5\beta}$				$(q_6,*,R)$					
q_6			(q_7,A,R)	(q_7,B,R)			$(q_f,*,R)$		
q_7			(q_7,A,R)	(q_7,B,R)	$(q_{8\alpha},*,L)$	$(q_{8\beta},*,L)$	$(q_7,*,R)$		
$q_{8\alpha}$			$(q_{4\alpha},A,L)$	$(q_{4\alpha},B,L)$			$(q_{8\alpha},*,L)$		
$q_{8\beta}$			$(q_{4\beta},A,L)$	$(q_{4\beta},B,L)$			$(q_{8\beta},*,L)$		
q_f									

q_0: On commence sur >, on va à droite, on marque la première lettre, et on va en q_1;
q_1: On va jusqu'à la fin du mot, puis on va en q_2;
q_2: On est sur la dernière lettre, que l'on marque, et on va en q_3;
q_3: On revient au début, jusqu'à rencontrer la première lettre marquée; on retourne alors en $q_{0;}$
q_0: On marque la deuxième lettre, et on va en q_1;
q_1: On va jusqu'à la première lettre marquée, et on va en q_2;

q_2: On est sur la dernière lettre non marquée, que l'on marque, et on va en q_3;
q_3: On revient au début, jusqu'à rencontrer la première lettre marquée; on retourne alors en q_0;
On répète ainsi la séquence q_0, q_1, q_2, q_3, jusqu'à ce que, en arrivant en q_0, il n'y ait plus de lettres non marquées. On tombe donc sur un α ou un β : on peut alors commencer la comparaison des deux demi-mots. On remplace la lettre lue par une étoile, et on la mémorise en allant soit en $q_{4\alpha}$ soit en $q_{4\beta}$.
$q_{4\alpha}$ ou $q_{4\beta}$: On va jusqu'au début du mot, puis on va en $q_{5\alpha}$ ou $q_{5\beta}$;
$q_{5\alpha}$ ou $q_{5\beta}$: On est sur la première lettre marquée; celle-ci doit être un *A* si on est en $q_{5\alpha}$ et un *B* si on est en $q_{5\beta}$; si tel est le cas on remplace la lettre lue par une étoile, et on va en q_6;
q_6: On teste si l'on n'est pas sur une étoile; si tel est le cas, c'est que l'on a comparé avec succès toutes les lettres des deux demi-mots, on a donc fini, on va en q_f; sinon on va en q_7;
q_7: On remonte le mot jusqu'à la première lettre α ou β que l'on mémorise en allant en $q_{8\alpha}$ ou $q_{8\beta}$;
$q_{8\alpha}$ ou $q_{8\beta}$: on redescend les étoiles, jusqu'à retomber sur un *A* ou un *B*, et on continue la descente en allant en $q_{4\alpha}$ ou $q_{4\beta}$;