SÉRIE D'EXERCICES n° 2

EXERCICE **1** : Pour chacun des langages suivants, construire un automate d'états finis qui l'accepte :

a) $L_1 = \{ w \in \{a, b, c\}^* \,/\, w$ contient au moins une occurrence de la lettre 'a' $\}$;

b) $L_2 = \{ w \in \{a, b\}^* \,/\, w = a^n b^m a$ ou $w = ba^n$; $n, m \geq 1 \}$;

c) $L_3 = \{ w \in \{0, 1\}^* \,/\, w = 1(101)^n 00$ ou $w = 0(010)^n 11, n \geq 0 \}$;

d) $L_4 = \{ w \in \{a, b\}^* \,/\, |w|$ est divisible par 6 $\}$;

e) $L_5 = \{ w \in \{0, 1\}^* \,/\, w$ ne contient pas la sous chaîne «010» $\}$;

f) L_6 = ensemble des mots de $\{0, 1\}^*$ représentant les nombres divisibles par 3 (dans le système de numération binaire naturel).

EXERCICE **2** : Etant donné les automates d'états finis non déterministes :

a) $A = < V , S , F , S_0 , I >$ où $V = \{0, 1\}$; $S = \{S_0 , S_1 , S_2\}$; $F = \{S_1 , S_2\}$; S_0 état initial
$I = \{ (0 , S_0 , S_0) ; (0 , S_0 , S_1) ; (0 , S_1 , S_1) ; (1 , S_1 , S_2) ; (1 , S_0 , S_2) ; (1 , S_2 , S_2) ; (1 , S_2 , S_0) \}$;

b) $B = < V , S , F , S_0 , I >$ où $V = \{a, b\}$; $S = \{S_0 , S_1 , S_2 , S_3\}$; $F = \{S_1\}$; S_0 état initial
$I = \{ (a , S_0 , S_0) ; (a , S_0 , S_1) ; (b , S_0 , S_2) ; (b , S_1 , S_2) ; (a , S_1 , S_3) ; (a , S_2 , S_2) ; (b , S_2 , S_1) ;$
$(b , S_2 , S_3) ; (a , S_3 , S_1) ; (b , S_3 , S_2) \}$;

1) Dessiner le diagramme graphique représentant chacun des automates A et B.

2) Déterminer une grammaire régulière à droite équivalente à chacun des automates A et B.

3) Trouver un automate d'états finis déterministe équivalent à A et à B.

EXERCICE **3** : Considérons l'automate d'états finis généralisé suivant :

$A = < V , S , F , S_0 , I >$ où $V = \{a, b\}$; $S = \{S_0 , S_1 , S_2, S_3 , S_4 , S_5\}$; $F = \{S_5\}$; S_0 état initial
$I = \{ (\varepsilon , S_0 , S_1) ; (\varepsilon , S_0 , S_3) ; (a , S_1 , S_2) ; (ab , S_2 , S_2) ; (a , S_2 , S_5) ; (b , S_3 , S_3) ; (a , S_3 , S_4) ;$
$(a , S_4 , S_5) \}$;

Construire l'automate simple équivalent à A.

EXERCICE **4** : Considérons l'automate d'états finis généralisé suivant :

$A = < V , S , F , S_0 , I >$ où $V = \{0,1\}$; $S = \{S_0 , S_1 , S_2, S_3\}$; $F = \{S_3\}$; S_0 état initial
$I = \{ (00 , S_0 , S_1) ; (0 , S_0 , S_2) ; (\varepsilon , S_1 , S_0) ; (\varepsilon , S_1 , S_3) ; (\varepsilon , S_2 , S_1) ; (1 , S_2 , S_3) ; (0 , S_3 , S_3) \}$;

1) Construire l'automate simple et déterministe équivalent à A.

2) Construire l'automate qui accepte le complémentaire de L(A).

EXERCICE **5** : Pour chacune des grammaires g1 et g2,

1) construire l'automate simple déterministe qui accepte $L(g_i)$, i=1,2 ;

2) construire l'automate simple déterministe qui accepte le complémentaire de $L(g_i)$, i=1,2 ;

grammaire g1 = $<\pi, N_1, S, P_1>$ où $\pi = \{a, b, c\}$; $N_1 = \{S, A, B\}$;

$$P_1 = \{ S \rightarrow abS \mid bA \mid \varepsilon ;\; A \rightarrow bB \mid B ;\; B \rightarrow aB \mid b \mid cA \}\ ;$$

grammaire g2 = <π, N_2, S, P_2> où π = {a, b, c} ; N_2 = {S, A} ;

$P_2 = \{ S \rightarrow aS \mid bS \mid cS \mid abcA ;\ A \rightarrow aA \mid bA \mid cA \mid \varepsilon \}$.

EXERCICE **6** :

Soit L_1 le langage des mots de $\{a, b\}^*$ contenant un nombre impair de lettres «a» ; et $L_2 = \{aa, ab\}$.

1) Construire un automate d'états finis simple qui accepte L_1.
2) Construire un automate d'états finis simple qui accepte L_2.
3) Construire un automate d'états finis simple qui accepte $L_1 \cup L_2$.
4) Rendre l'automate de 3) déterministe.

EXERCICE **7** : Soit la grammaire g = <{a, b}, {A, B, C}, A, P> où

$P = \{ A \rightarrow aaB \mid abC ;\ B \rightarrow aC \mid a ;\ C \rightarrow bC \mid \varepsilon \}$

1) Construire l'automate simple déterministe qui accepte l'itération de L(g).
2) Construire l'automate qui accepte le reflet miroir de L(g).

EXERCICE **8** : Soit L_1 et L_2 deux langages définis sur V = {a, b} par :

$L_1 = \{ w \in V^* / w$ ne contient pas de sous mot «ab» $\}$; $L_2 = \{ (ba)^n.b / n \geq 0 \}$

1) Trouver une grammaire régulière à droite pour chacun des langages L_1 et L_2.
2) A partir des grammaires trouvées en 1), construire un automate d'états finis pour chacun des langages L_1 et L_2.
3) Trouver un automate d'états finis qui accepte le langage $L_1.L_2^*$.

EXERCICE **9** : Soit V un alphabet fini ; et soient A = <V, S_1, F_1, S_{01}, I_1> l'automate qui accepte un langage L, et B = <V, S_2, F_2, S_{02}, I_2> l'automate qui accepte le langage L'.

On supposera que A et B sont déterministes et complets.

On construit l'automate C = <V, S, F, S_0, I> qui accepte le langage L'' = L ∩ L' comme suit :

$S = S_1 \times S_2$, $F = F_1 \times F_2$, $S_0 = (S_{01},S_{02})$ et I est tel que :

$(a , (S_i,S_j) , (S_k,S_l)) \in I$ si et seulement si $(a , S_i , S_k) \in I_1$ et $(a , S_j , S_l) \in I_2$.

Construire l'automate C pour L = ensemble des chaînes de 0 et 1 représentant les nombres pairs (dans le système de numération binaire naturel) et L' = langage L_6 de l'exercice **1** de cette série.

EXERCICE **10** : (*exercice à programmer*)

On veut représenter un automate d'états finis simple déterministe par une structure de données.

Pour cela on utilise la table de transitions de l'automate.

1) Ecrire un algorithme puis un programme Pascal permettant de tester si une chaîne donnée appartient au langage accepté par l'automate.
2) Reconsidérer la question 1) pour un automate non déterministe.

EXERCICE **11** : (*exercice à programmer*)

On veut représenter un automate d'états finis simple déterministe par un programme Pascal.

Dans ce cas on associera des étiquettes (*label*), aux états de l'automate, dans le programme.

On pourra prendre comme exemple l'automate de f) de l'exercice **1**.