Université Mouloud MAMMERI de Tizi-Ouzou
Faculté de génie électrique et informatique
Département d'informatique

Année universitaire : 2017/2018
2ième année Licence-Informatique
module : Théorie des langages

CORRIGÉ ABRÉGÉ DE LA SÉRIE D'EXERCICES n° 1 de ThL

par : **M.S. Habet, C. Cherifi, F. Bouhatem, N. Otmani**

EXERCICE **1** :

1) x = acabacbc
2) $|x| = 8$, $|x|_a=3$, $|x|_b=2$, et $|x|_c=3$
3) acabac
4) acbc

EXERCICE **2** :

Soit V un alphabet.

- On montre que pour tous u, v de V^*, on a : $(u.v)^R = v^R.u^R$ (I)

Soient u et v deux mots de V^*, et $n = |u|$. On raisonne par récurrence sur n.

-- pour n=0 :

dans ce cas $u = \varepsilon$ et on a bien $(u.v)^R = (\varepsilon.v)^R = v^R = v^R.\varepsilon = v^R.\varepsilon^R = v^R.u^R$

d'où (I) est vérifiée.

Supposons que (I) reste vraie jusqu'au rang n=k. (Ir)

-- pour n=k+1 :

dans ce cas il existe u' dans V^* et a dans V tels que u = a.u' (on a $|u'| = k$).

$(u.v)^R = ((a.u').v)^R = (a.(u'.v))^R$ (d'après l'associativité de la concaténation)

$= (u'.v)^R.a$ (d'après la définition du reflet miroir)

$= v^R.u'^R.a$ (d'après l'hypothèse (Ir))

$= v^R.u^R$ (d'après la définition du reflet miroir)

D'où le résultat : (I) est toujours valide.

- On montre que pour tout w de V^*, on a : $(w^R)^R = w$ (II)

Soit w un mot de V^*, et $n = |w|$. On raisonne par récurrence sur n.

-- pour n=0 :

$w = \varepsilon$, par définition $w^R = \varepsilon$ et aussi : $(w^R)^R = \varepsilon = w$.

(II) est vérifiée.

Supposons que (II) reste vraie jusqu'au rang n=k. (IIr)

-- pour n=k+1 :

Dans ce cas, il existe $a \in V$ et $v \in V^*$ tels que w = a.v, et : $w^R = v^R.a$.

On aura : $(w^R)^R = (v^R.a)^R = a^R.(v^R)^R = a.(v^R)^R$ (d'après la formule (I))

$= a.v = w$ (d'après l'hypothèse de récurrence).

D'où le résultat : (II) est toujours valide.

EXERCICE **3** :

a) $L_1.L_2$ = {a, ab, aab, abb, abab, ba, bab, baab } ;
b) $L_2.L_1$ = {a, ab, ba, bab, bba, aba, abab, abba } ;

c) $L_1.L_3 = \{a^{n+1}.b^n \mid n \geq 0\} \cup \{a.b.a^n.b^n \mid n \geq 0\} \cup \{b.a^{n+1}.b^n \mid n \geq 0\}$;

d) et e) $L_1.\{\varepsilon\} = \{\varepsilon\}.L_1 = L_1$;

f) et g) $L_1.\varnothing = \varnothing.L_1 = \varnothing$;

h) $L_1.L_1 = \{aa, aab, aba, abab, abba, baa, baab, baba\}$;

i) $L_2.L_2 = \{\varepsilon, b, ab, bb, bab, abb, abab\}$;

j) $L_3.L_3 = \{a^n.b^n.a^m.b^m \mid n, m \geq 0\}$.

EXERCICE **4** :

1) Les mot w_1 et w_3 ne sont pas générés par G ;
les mots w_2 et w_4 sont générés par G : $S \vdash aS \vdash aaS \vdash aabA \vdash aabcA \vdash aabccA \vdash aabcccA \vdash w_2$
et pour w_4 : $S \vdash aS \vdash abA \vdash ab = w_4$.

2) Soit $L = \{ a^i b c^j \mid i, j \geq 0 \}$. Montrons que L(G)=L en prouvant la double inclusion :

- $L(G) \subseteq L$: soit w un mot de L(G), donc w est généré à partir de S en appliquant n fois les règles de production de G. Montrons par récurrence sur n que $w \in L$:
  - si n=2 alors on a : $S \vdash bA \vdash b$; $w=b \in L$. Supposons que la propriété reste vraie jusqu'au rang n=k.
  - pour n=k+1, on a deux cas :
    -- la première règle appliquée est $S \to aS$, puis k règles pour avoir un mot a.u. Puisque u est généré à partir de S avec application de k règles de G, et d'après l'hypothèse de récurrence, u est dans L, donc il s'écrit comme $u = a^i b c^j$ et ainsi le mot $a.u = a^{i+1} b c^j \in L$.
    -- la première règle appliquée est $S \to bA$, puis à partir de A, on obtient c^j ($j \geq 0$), et on aura donc généré le mot $b.c^j$ qui $\in L$ (c'est : $a^i.b.c^j$ avec i=0).
- $L \subseteq L(G)$: Soit $w \in L$. Donc w s'écrit comme $w = a^n b c^m$. w peut être dérivé de S en appliquant n fois la règle $S \to aS$ puis une fois la règle $S \to bA$, puis encore m fois la règle $A \to cA$ et enfin une fois la règle $A \to \varepsilon$. Donc $w \in L(G)$.

EXERCICE **5** :

1) Exemples de mots de L_1 : a, ab, accb, …
Une grammaire pour L_1 : $G_1 = (\{a, b, c\}, \{S, A\}, S, \{ S \to aA ; A \to aA \mid bA \mid cA \mid \varepsilon \})$

2) Exemples de mots de L_2 : a, ba, cabca, …
Une grammaire pour L_2 : $G_2 = (\{a, b, c\}, \{S\}, S, \{ S \to aS \mid bS \mid cS \mid a \})$

3) Exemples de mots de L_3 : a, bba, acaacb, …
Une grammaire pour L_3 : $G_3 = (\{a, b, c\}, \{S, A\}, S, \{ S \to aS \mid bS \mid cS \mid aA ; A \to aA \mid bA \mid cA \mid \varepsilon \})$

4) Exemples de mots de L_4 : aa, aba, acacab, …
Une grammaire pour L_4 : $G_4 = (\{a, b, c\}, \{S, A, B\}, S, P_4)$
P_4 : $\{ S \to aS \mid bS \mid cS \mid aA ; A \to aA \mid bA \mid cA \mid aB ; B \to aB \mid bB \mid cB \mid \varepsilon \})$

5) Exemples de mots de L_5 : aa, baab, accaab, …
Une grammaire pour L_5 : $G_5 = (\{a, b, c\}, \{S, A\}, S, \{ S \to aS \mid bS \mid cS \mid aaA ; A \to aA \mid bA \mid cA \mid \varepsilon \})$

EXERCICE **6** :

I) Nous donnons ici les types des G_i, (i=1,..,6), ainsi que les langages engendrés par les grammaires G_i (i=1,..,6). (Pour que la réponse soit complète, il faut le prouver comme c'est fait dans l'exercice 4).

1) Type de G_1 = 3. $L(G_1) = \{ aa, aab, bb, bcb \}$.

2) Type de G_2 = 3. $L(G_2) = \{ b.a^n \ / \ n \geq 0 \}$.

3) Type de G_3 = 2. $L(G_3) = \{ a^n b^m c^n \ / \ n \geq 0, m \geq 1 \}$.

4) Type de G_4 = 2. $L(G_4) = \{ w \in \{a, b\}^* \ / \ |w|_a = |w|_b$ et $\forall$ u préfixe de w, $|u|_a \geq |u|_b \}$.

5) Type de G_5 = 1. $L(G_5) = \{ a^n b^n c^n \ / \ n \geq 1 \}$.

6) Type de G_6 = 0. $L(G_6) = \{ a^n b^{2.[n/2]} \ / \ n \geq 0 \}$; ([x] est la partie entière de x)
   On peut aussi écrire $L(G_6)$ comme $\{ a^{2k+1} b^{2k} \ / \ k \geq 0 \} \cup \{ a^{2k} b^{2k} \ / \ k \geq 0 \}$.

II) G_2 n'est pas de type 1 car elle contient la règle : $A \rightarrow \varepsilon$; or dans les grammaires de type 1, le seul symbole qui peut produire la chaîne vide est S.
Cependant, on peut écrire une grammaire de type 1 équivalente à G_2 :
G_2' a pour règles de production : $S \rightarrow Sa \mid b$; ce qui veut dire que $L(G_2)$ est de type 1.

III) Une grammaire de type 2 équivalente à G_6 :
G_6' a pour règles de production : $S \rightarrow aaSbb \mid a \mid \varepsilon$

EXERCICE **7** :

a) pour L_1 : il est engendré par $G_1 = (\{0\}, \{S\}, S, P_1)$, où P_1 : $S \rightarrow 00S \mid \varepsilon$

b) pour L_2 : il est engendré par $G_2 = (\{0, 1\}, \{S\}, S, P_2)$, où P_2 : $S \rightarrow 0S1 \mid \varepsilon$

c) pour L_3 : il est engendré par $G_3 = (\{a, b\}, \{S\}, S, P_3)$, où P_3 : $S \rightarrow aSbb \mid \varepsilon$

d) pour L_4 : il est engendré par $G_4 = (\{a, b\}, \{S, A\}, S, P_4)$,
   où P_4 : $S \rightarrow aSc \mid A$; $A \rightarrow aAb \mid \varepsilon$

e) pour L_5 : il est engendré par $G_5 = (\{a, b\}, \{S\}, S, P_5)$,
   où P_5 : $S \rightarrow aSa \mid bSb \mid a \mid b \mid \varepsilon$

f) pour L_6 : il est engendré par $G_6 = (\{a, b\}, \{S, A\}, S, P_6)$
   où P_6 : $S \rightarrow aSa \mid bSb \mid aAb \mid bAa$
   $A \rightarrow aA \mid bA \mid \varepsilon$

g) pour L_7 : il est engendré par $G_7 = (\{a, b\}, \{S, A\}, S, P_7)$,
   où P_7 : $S \rightarrow AAAS \mid \varepsilon$;
   $A \rightarrow a \mid b$

h) pour L_8 : il est engendré par $G_8 = (\{0, 1\}, \{S\}, S, P_8)$,
   où P_8 : $S \rightarrow 0S1 \mid 0S \mid \varepsilon$

i) $L_9 = \{ 0^i 1^j /\ i > j \} \cup \{ 0^i 1^j /\ i < j \}$; L_9 est engendré par $G_9 = (\{0, 1\}, \{S, S_0, S_1\}, S, P_9)$,

où P_9 : $S \rightarrow S_0 \mid S_1$;

$S_0 \rightarrow 0S_01 \mid 0S_0 \mid 0$;

$S_1 \rightarrow 0S_11 \mid S_11 \mid 1$

j) L_{10} : il est engendré par $G_{10} = (\{a\}, \{S, A, B, C, D\}, S, P_{10})$,

où P_{10} : $S \rightarrow BCD$

$C \rightarrow AC \mid a$

$Aa \rightarrow aaA$

$AD \rightarrow D$

$Ba \rightarrow aB$

$BD \rightarrow \varepsilon$

EXERCICE **8** :

1) Soient les langage $L = \{0, 1\}^*$ et $L' = \{ 0^n1^n /\ n \geq 0 \}$. L est de type 3 (vérifier le !) ; mais L', qui est inclus dans L, n'est pas de type 3 (il est de type 2).
2) On considérera le cas d'une grammaire régulière à droite (l'autre cas à gauche est aussi vérifié).
   Toute grammaire régulière à droite respecte la condition que toute règle est de la forme $A \rightarrow wB \mid w$; où A est un non terminal, w un mot quelconque de terminaux (éventuellement vide) et B non terminal. Ces contraintes respectent aussi les conditions du type 2 ; à savoir des règles de la forme $A \rightarrow w$; où A est un non terminal et w est, dans ce cas, une suite quelconque (éventuellement vide) de terminaux et/ou de non terminaux. Donc toute grammaire régulière est à contexte libre.
   Mais la réciproque est fausse ; soit la grammaire à contexte libre (type 2) :
   $G_2 = (\{a, b\}, \{S\}, S, \{ S \rightarrow aSb \mid \varepsilon \})$. Elle vérifie les conditions du type 2 mais pas celles du type 3.

EXERCICE **9** :

1) L peut être généré par la grammaire, de type 3, $G = (\{a, b, c\}, \{S, C\}, S, P)$
   où P : $S \rightarrow aaS \mid bcC$
   $C \rightarrow ccC \mid \varepsilon$
2) Une autre grammaire de type 2, et qui n'est pas de type 3, qui engendre L :
   $G' = (\{a, b, c\}, \{S, A, C\}, S, P')$
   où P' : $S \rightarrow AbcC$
   $A \rightarrow aaA \mid \varepsilon$
   $C \rightarrow ccC \mid \varepsilon$

EXERCICE **10** :

1) L peut être généré par la grammaire, de type 3, $G = (\{0, 1\}, \{S, A\}, S, P)$
   où P : $S \rightarrow 0S \mid 1A \mid \varepsilon$
   $A \rightarrow 0A \mid 1S$
2) Une autre grammaire de type 2, et qui n'est pas de type 3, qui engendre L :
   $G' = (\{0, 1\}, \{S\}, S, P')$
   où P' : $S \rightarrow 0S \mid S1S1S \mid \varepsilon$

EXERCICE **11** :

1) $L(G) = \{ a^n b^m \ / \ n \leq m \leq 2*n \}$

2) Grammaire à contexte libre équivalente à G : G' = ({a, b}, {S, B}, S, P')

P' : $S \rightarrow aSbB \mid \varepsilon$

$B \rightarrow b \mid \varepsilon$

EXERCICE **12** :

1) $L(G) = \{ a^{2.i+1}.b^{2.j+1} \ / \ i, j \geq 0 \} \cup \{ a^{2.i+1}.b^{2.j} \ / \ i, j \geq 0 \} \cup \{ a^{2.i}.b^{2.j+1} \ / \ i, j \geq 0 \} \cup \{ a^{2.i}.b^{2.j} \ / \ i, j \geq 0 \}$

$= \{ a^n.b^m \ / \ n, m \geq 0 \}$.

2) Grammaire G' équivalente à G ayant moins de règles de production et qui a un seul symbole non terminal (S) : G' = ({a, b}, {S}, S, P')

où P' : $S \rightarrow aS \mid Sb \mid \varepsilon$

3) G' n'est pas de type 3. Voici une grammaire de type 3 équivalente à G' : G" = ({a, b}, {S, A}, S, P")

où P" : $S \rightarrow aS \mid A$

$A \rightarrow bA \mid \varepsilon$

EXERCICE **13** :

1) Une grammaire de type 3 pour L pourrait être G = ({a, +, *}, {S}, S, P)

P : $S \rightarrow a{+}S \mid a{*}S \mid a$

2) Une grammaire de type 2 pour L' pourrait être G' = ({a, +, *, (,)}, {S}, S, P')

P' : $S \rightarrow (S) \mid S{+}S \mid S{*}S \mid a$

EXERCICE **14** :

Pour générer ces identificateurs on utilisera la grammaire G = (π, N, <Id1>, P) ;

où π = {A..Z, a..z, 0..9} ; N = {<Id1>, <Id2>, <Id3>, <Lettre>, <Chiffre>}

et P : <Id1> → <Lettre> <Id2>

<Id2> → <Id3> <Id2> | ε

<Id3> → <Lettre> | <Chiffre>

<Lettre> → A | B | .. | Z | a | b | .. | z

<Chiffre> → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

------------------------ Fin du corrigé de la série n° 1 de ThL ------------------------