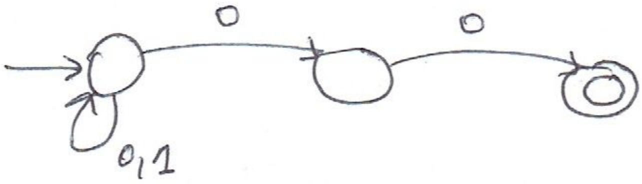
Série N° 1

exercice 01 :

1) Toutes les chaines qui se terminent par 00 .



$(0+1)^*00$

Automate Non déterministe

2) $(0+1)^*1\underset{9}{(0+1)}\underset{8}{(0+1)}\underset{7}{(0+1)}\underset{6}{(0+1)}\underset{5}{(0+1)}\underset{4}{(0+1)}\underset{3}{(0+1)}\underset{2}{(0+1)}\underset{1}{(0+1)}$

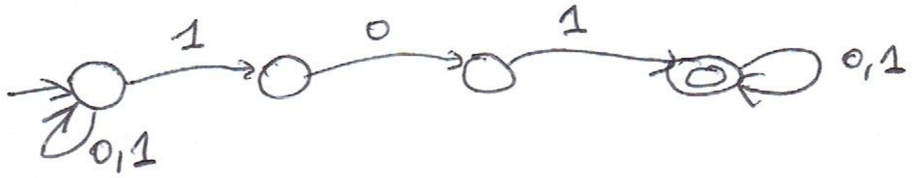
Toutes les chaînes dont le 10^{ieme} symbole, compté à partir de la fin de la chaine, est un 1

3) Toutes les chaines dans lesquelles chaque paire de 0 apparait devant une paire de 1.

$(1100+011+01+1)^*$

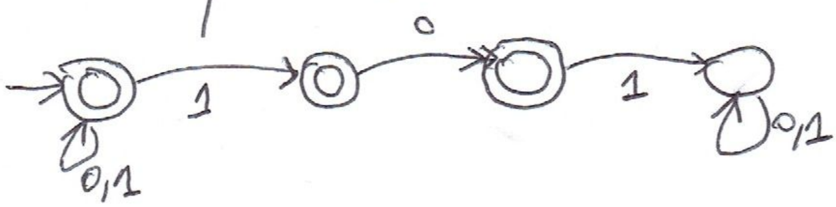
4) Toutes les chaines ne contenant pas 101 .

4.a)



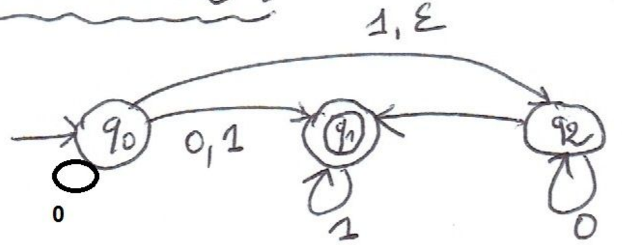
Le complément

4.b)



$(0+1)^* + (0+1)^*1 + (0+1)^*10$

Exercice 02 :



cet automate est non deterministe

001 est acceptée.

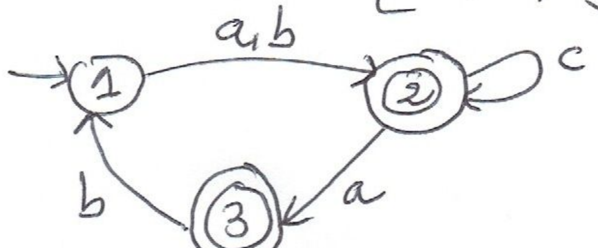
0^*11^* est acceptée

0^*110 est non acceptée.

Le langage L : $0^*(0+1)1^* + 0^*(1+\varepsilon)0^*11^*$

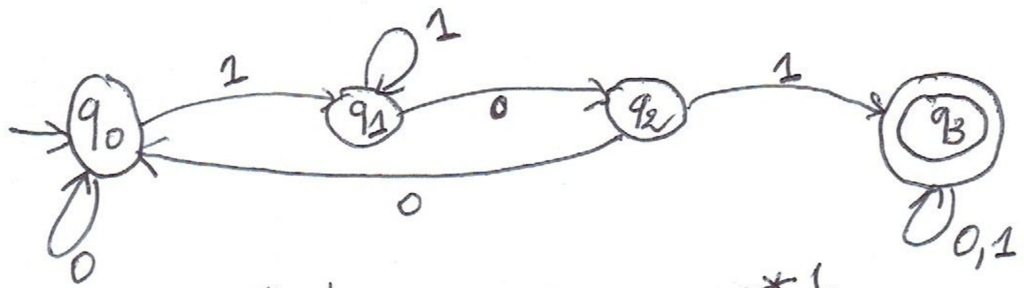
Les états : $\{1, 2, 3\}$

Les terminaux $\{a, b, c\}$

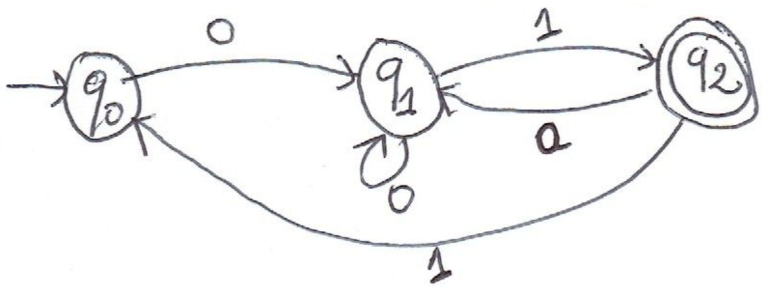


Exercice 03 :

$L_2 = \{x \in \Sigma^* \,/\, x$ contient la sous chaine $101\}$

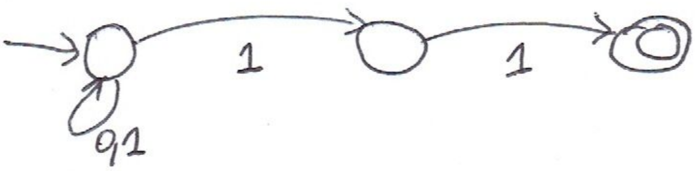


$L_1 = \{x \in \Sigma^* \mid x = y01 \,,\, y \in \Sigma^*\}$



Escercice 04 :

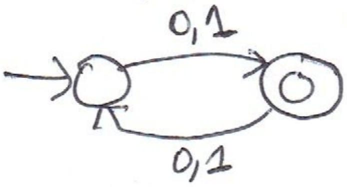
a) Les mots qui se terminent par 11.



- $(0+1)^* 11$
- Non déterministe.

b) Les mots de longueur impaire

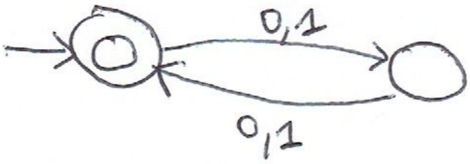
$L = \{ x \in \Sigma^* \,/\, |x| = 2k+1 \,/\, k \geq 0 \}$.



- $(0+1)\big((0+1)(0+1)\big)^*$
- déterministe

c) Les mots de longueur paire.

$L = \{ x \in \Sigma^* \,/\, |x| = 2k \,,\, k \geq 0 \}$

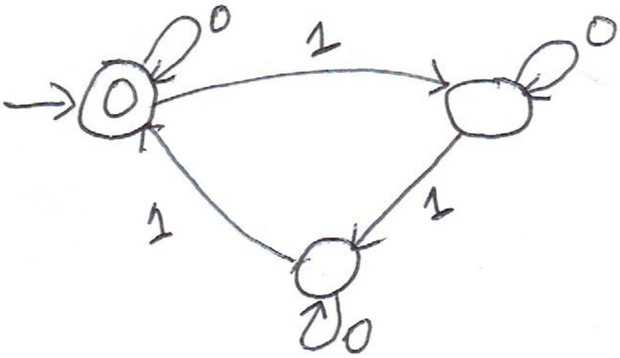


- déterministe
- $((0+1)(0+1))^*$

d) Les mots qui commencent et se terminent par 0

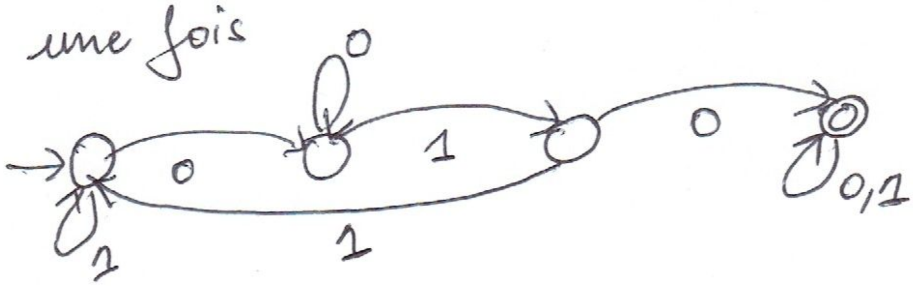
- $0(0+1)^* 0$
- Non déterministe

e) Les mots où le nb d'occurences des 1 est multiple de 3



- $(0^* 1\, 0^* 1\, 0^* 1)^*$
- déterministe

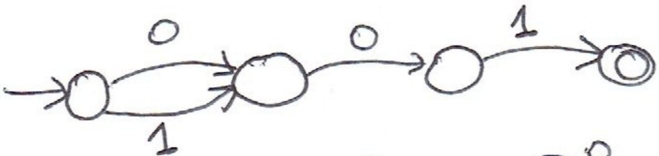
f) les mots qui contiennent la chaine 010 au moins une fois



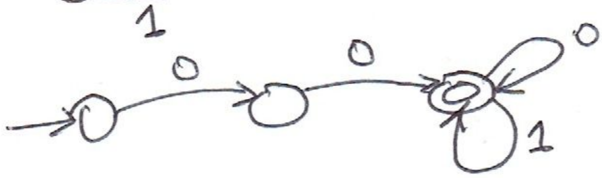
Exercice 5 :

$r_1 = 01^*$

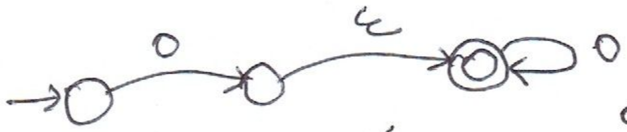
$r_2 = (0+1)01$



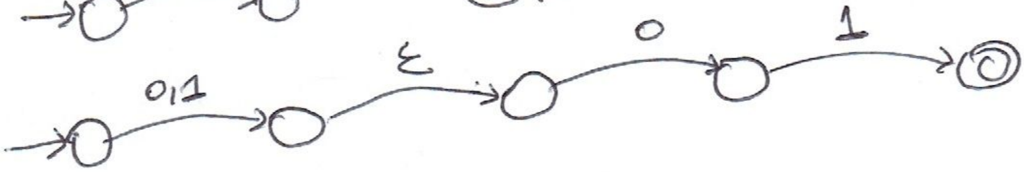
$r_3 = 00(0+1)^*$



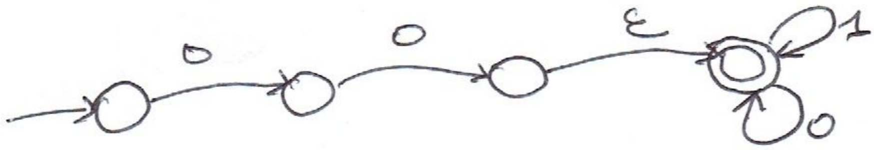
r_1 :



r_2 :

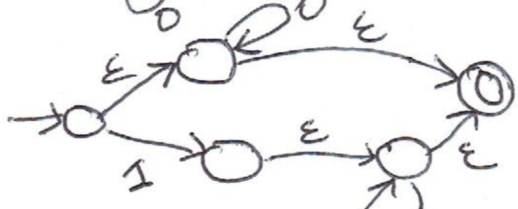


r_3 :

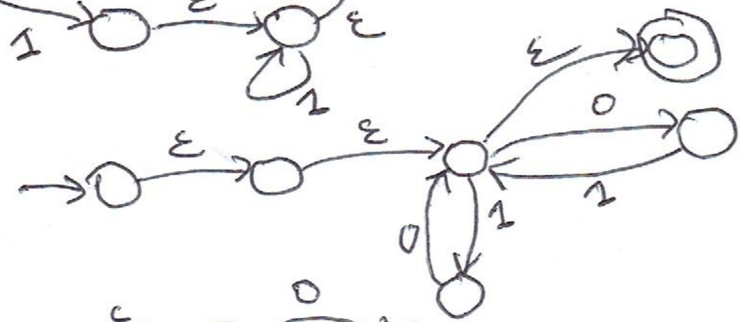


r_4 : $0^* 1^*$

r_5 : $0^* + 11^*$



r_6 : $0^*(01+10)^*$



$r_7 = (01^*)^*$

Exercice 6 :

L1 pour figure1

$L1 = \{\varepsilon, 10^* + 010^* + 0\}$

figure 2 :

$L2 = \{ \quad (01 + 010)^* \}$

figure 3 :

$L3 = \{\varepsilon + 01 + 010 + 01(01)^* + 010(010)^*\}$

$= \{\varepsilon + 01(\varepsilon + 01)^* + 010(\varepsilon + (010)^*)\}$

$= \{\varepsilon + 01[(\varepsilon + 01)^* + 0(\varepsilon + (010)^*)]\}$

$\boxed{L3 = \{\varepsilon + 01(\varepsilon + 0)(01 + 010)^*\}}$

figure 4 :

$L4 = \{\varepsilon + (001)^* + 01(11)^*\}$

figure 5 :

$L5 = \{(00)^* + 0(11)^*0 + 0(11)^*10(00)^*1(11)^* + 0(11)^* + \ldots\}$

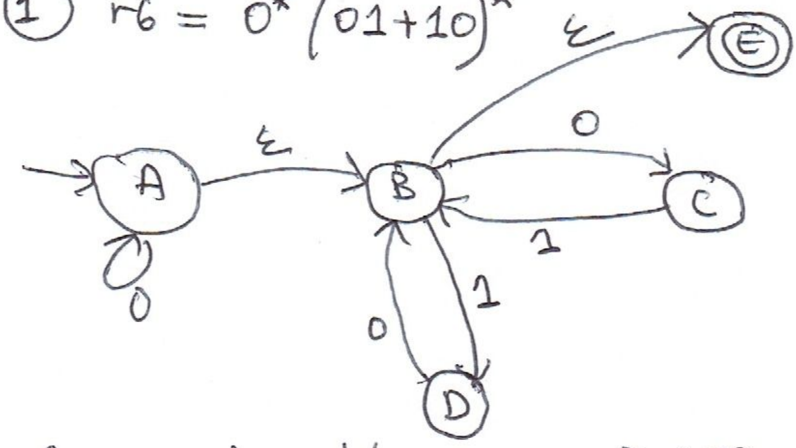
L5 se compose des mots qui contiennent un nb pair de 0 ou un nb pair de 1.

figure 6 :

$L6 = \{0 + 00 + 01 + 00(0+1)(0+1)^* + 01(0+1)(0+1)^* + 11^*0(0+1)^*$
$+ 10 + 10(0+1)(0+1)^*\}$

Exercice 7:

① $r6 = 0^*(01+10)^*$



Conversion d'1 AFND à AFD

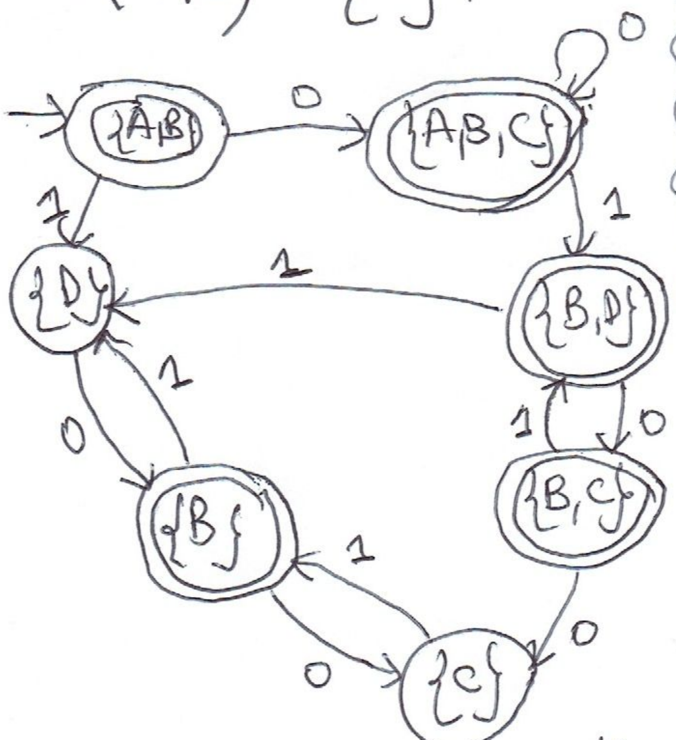
$\delta(A,0) = \{A,B\}$

$\delta(B,0) = \{C\}$

$\delta(B,1) = \{D\}$

$\delta(D,0) = \{B\}$

$\delta(D,\varepsilon) = \{E\}$.



L'automate deterministe equivalent

$\delta(A,0) = \{A,B\}$

$\delta(\{A,B\},0) = \{A,B,C\}$

$\delta(\{A,B\},1) = \{D\}$

$\delta(\{A,B,C\},0) = \{A,B,C\}$

$\delta(\{A,B,C\},1) = \{B,D\}$

$\delta(\{D\},0) = \{B\}$

$\delta(\{D\},1) = \emptyset$

$\delta(\{B,D\},0) = \{B,C\}$

$\delta(\{B,D\},1) = \{D\}$

$\delta(\{B\},0) = \{C\}$

$\delta(\{B\},1) = \{D\}$

$\delta(\{B,C\},0) = \{C\}$

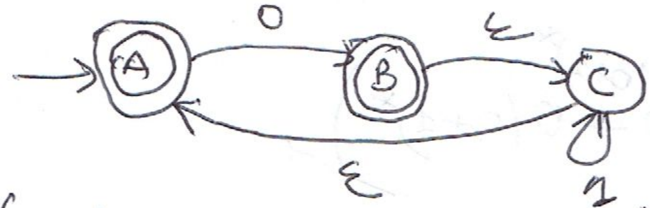
$\delta(\{B,C\},1) = \{B,D\}$

$\delta(\{C\},0) = \emptyset$

$\delta(\{C\},1) = \{B\}$

Suite de l'exo 7 :

$r7 = (0 1^*)^*$



$\delta(A,0) = \{A,B,C\}$

$\delta(A,1) = \phi$

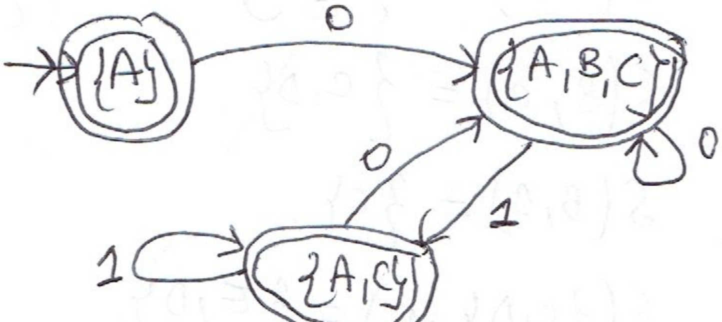
$\delta(\{A,B,C\},0) = \{A,B,C\}$

$\delta(\{A,B,C\},1) = \{A,C\}$

$\delta(\{A,C\},0) = \{A,B,C\}$

$\delta(\{A,C\},1) = \{A,C\}$

L'automate deterministe equivalent est :

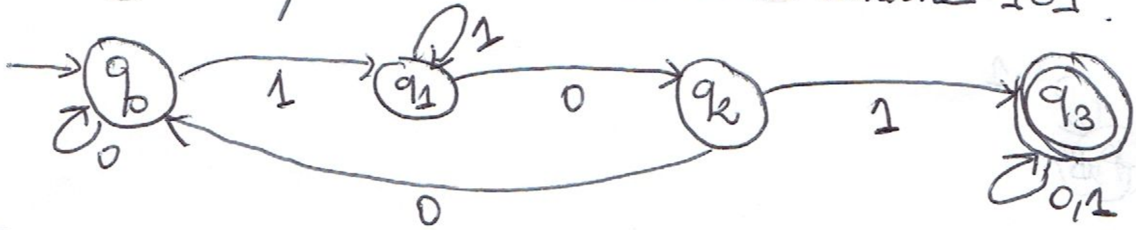


Exercice 8 :

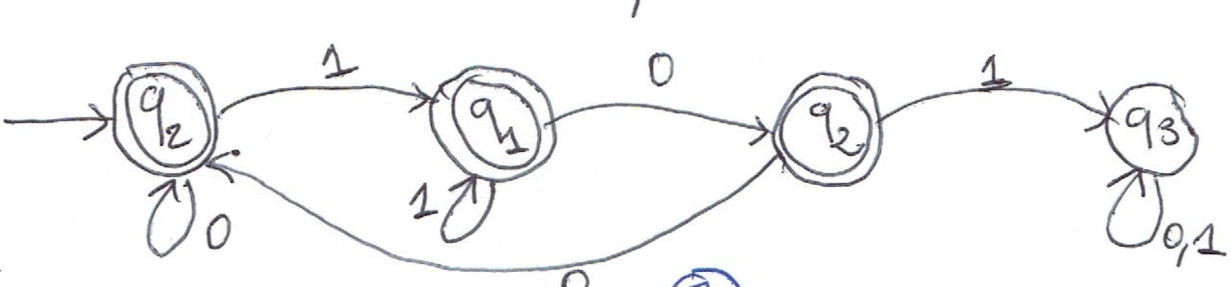
L'AFD M tel que $L(M) = \{x \in \Sigma^* / x$ ne contient pas la sous chaine $101\}$

Suite à l'exo3

$L2 = \{x \in \Sigma^* / x$ contient la sous-chaine 101

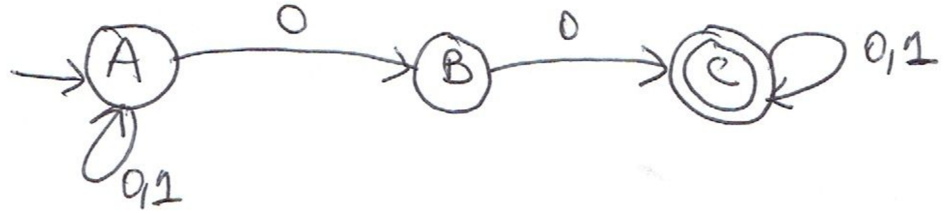

(A)

Le determinant le complément de A est $\overline{A}$

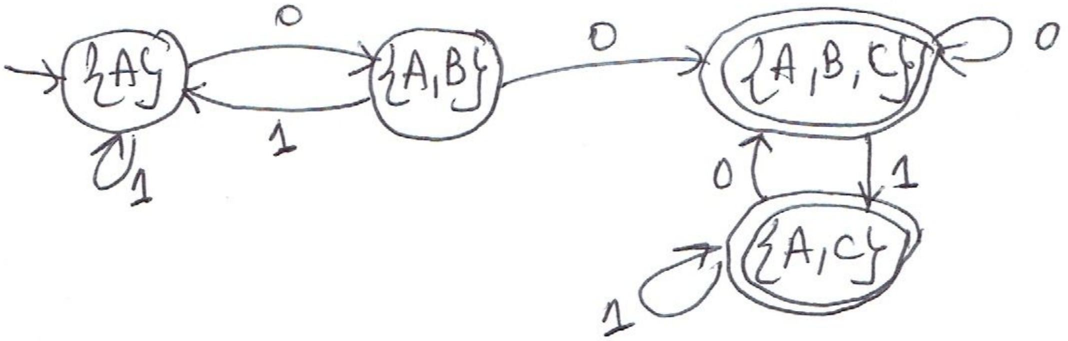


Exercice 09 :

① $L_1 = \{x \in \Sigma^* \,/\, x$ contient la sous chaine $00\}$

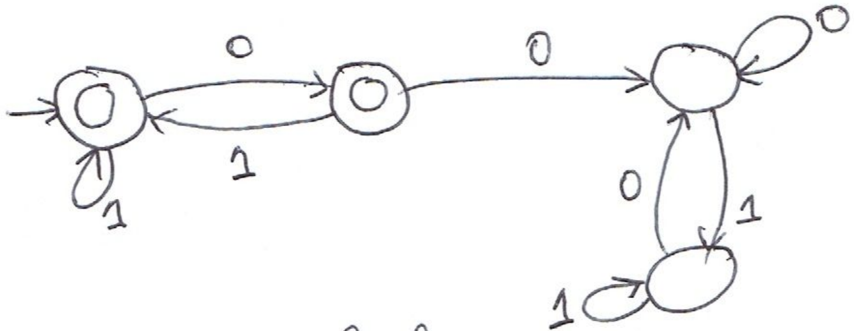


AFND



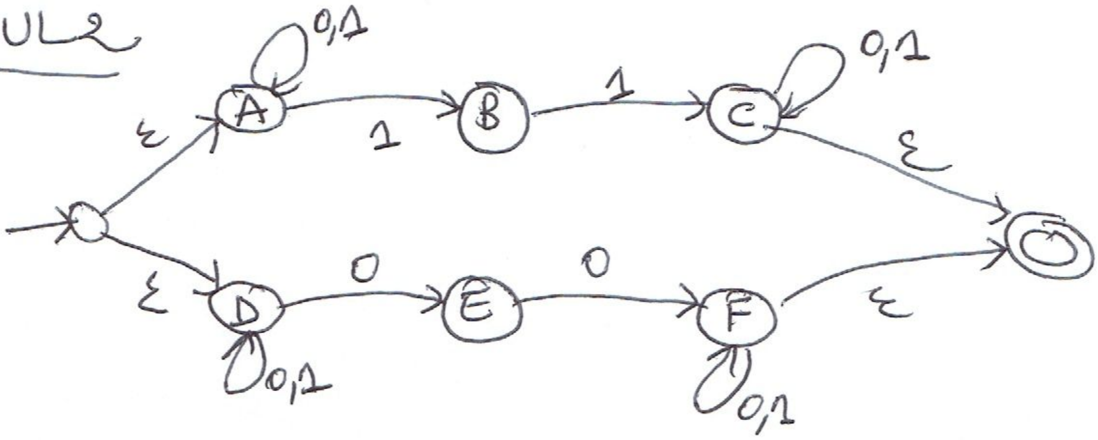
AFD

$\bar{L}_1 = \Sigma - L_1 \Leftrightarrow$

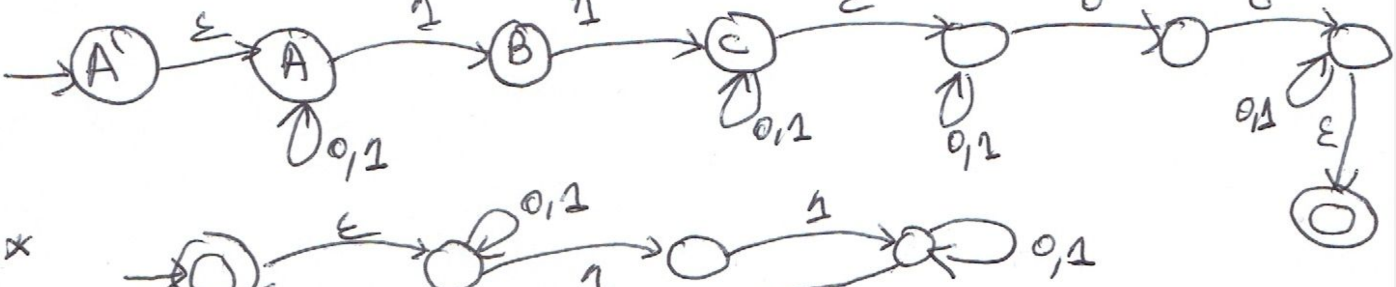


② AFND pour les langages $L_1 \cup L_2$, $L_1 L_2$, L_1^*, Σ_1

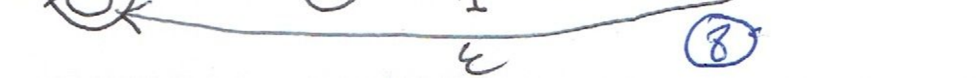
$L_1 \cup L_2$



L2.L1

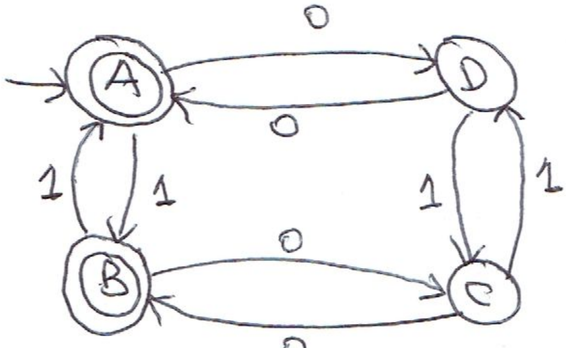


L_2^*



Exo 10 :

Figure 05



$M_0 = \{\{A,B\},\{C,D\}\}$

$A \overset{?}{\equiv} B \begin{cases} \delta(A,0) \equiv \delta(B,0) & D \equiv_0 C \checkmark \\ \delta(A,1) \equiv \delta(B,1) & B \equiv_1 A \checkmark \end{cases}$

$C \overset{?}{\equiv} D \begin{cases} \delta(C,0) \equiv \delta(D,0) ; & B \equiv_0 A \checkmark \\ \delta(C,1) \equiv \delta(D,1) ; & D \equiv_1 C \checkmark \end{cases}$

$M_1 = \{\{A,B\},\{C,D\}\} = M_0$.

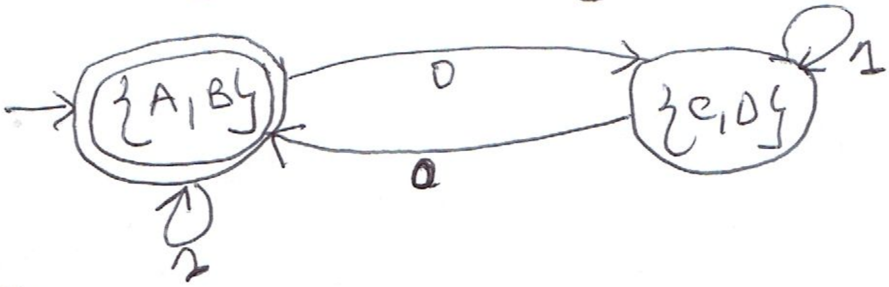
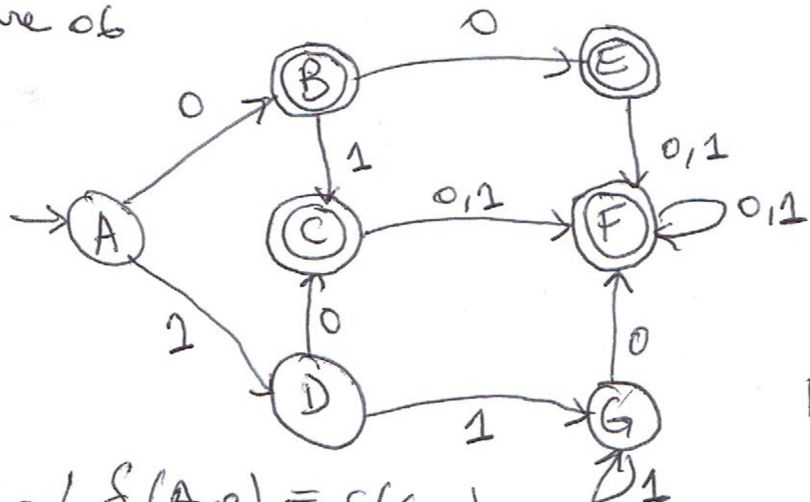


Figure 06



$M_0 = \{\{A,D,G\},\{B,C,E,F\}\}$

Les etats non terminaux

Les etats finaux

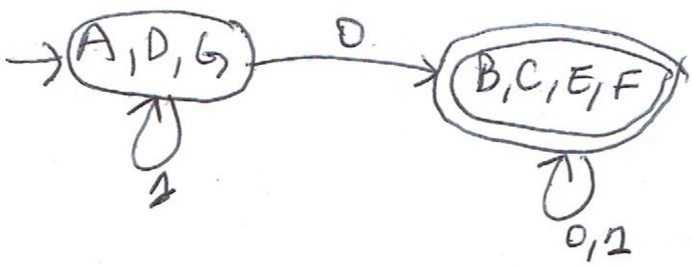
$A \equiv D \begin{cases} \delta(A,0) \equiv \delta(D,0) \\ \delta(A,1) \equiv \delta(D,1) \end{cases}$

$A \equiv G \begin{cases} \delta(A,0) \equiv \delta(G,0) \\ \delta(A,1) \equiv \delta(G,1) \end{cases}$

$D \equiv G \begin{cases} \delta(D,0) \equiv \delta(G,0) \\ \delta(D,1) \equiv \delta(G,1) \end{cases}$

$\Rightarrow A \equiv D \equiv G$

$B \equiv C \equiv E \equiv F$



Serie N° 2

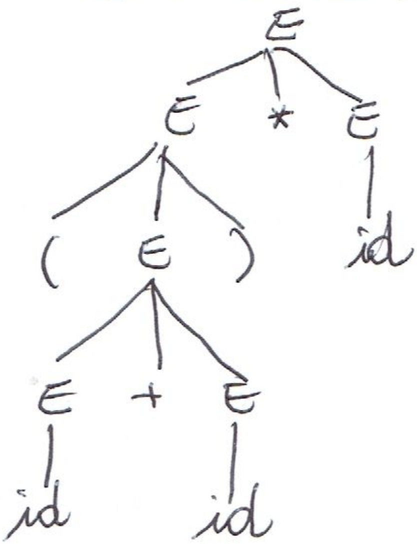
Exo1 :

Soit $G = (\{E\}, \{+, *, (,), id\}, P, E)$ tel que

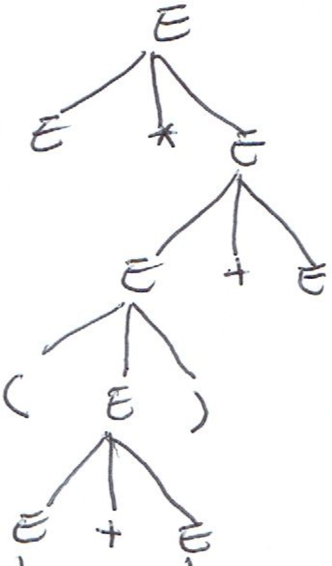
$P : E \to E+E \,/\, E*E \,/\, (E) \,/\, id$

① Les arbres de derivations

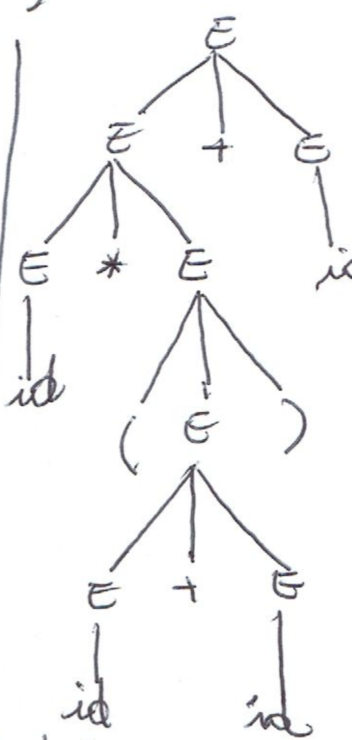
Pour le mot $(id+id)*id$



Pour le mot $id*(id+id)+id$



arbre1



arbre2

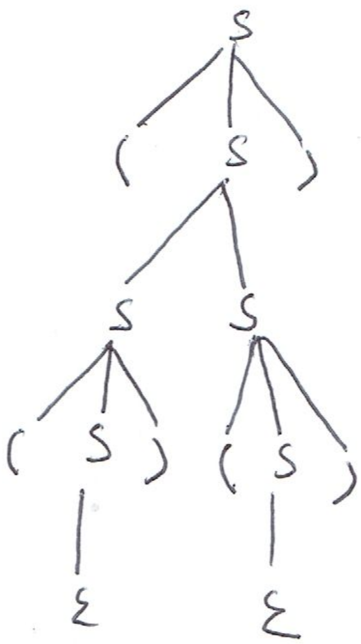
② Cette grammaire est ambigue car pour le mot $id*(id+id)+id$, il y a deux arbres de derivation différents

Exo2 :

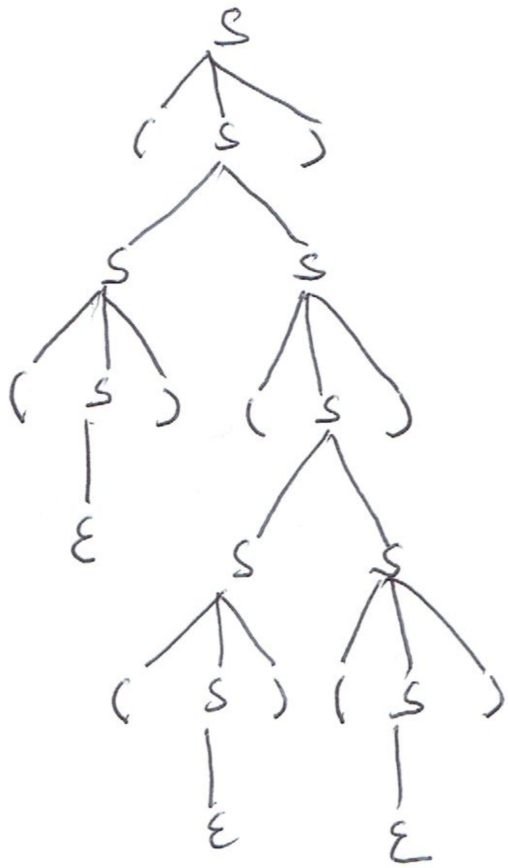
Soit $G = (\{S\}, \{0,1\}, P, S)$ où

$S \to SS \mid (S) \mid \varepsilon$

① L'arbre de dérivation pour le mot (()())



L'arbre de derivation pour (()(()()))



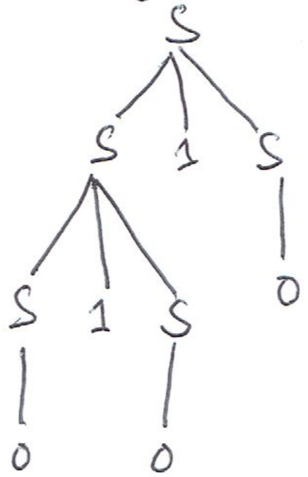
② Le langage généré par G est un langage bien parenthesé.

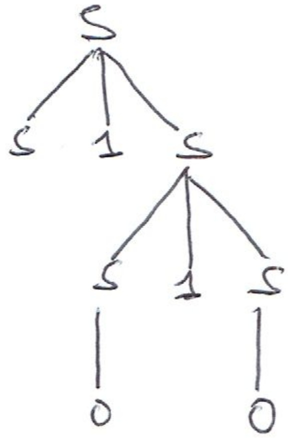
Exo 3 :

$G = (\{S\}, \{0,1\}, P, S)$ où $S \rightarrow S1S \mid 0$

1) On prend le mot 01010

Il y a deux arbres de dérivation différents





$\Rightarrow$ cette grammaire est ambigue.

2) $\underline{S} \rightarrow \underline{S}1S \rightarrow \underline{S}1S1S \rightarrow \underline{S}1S1S1S \rightarrow S1S1S1S1S \rightarrow$

$\cdots\cdots \rightarrow 01010101S \rightarrow 010101010101010 \overset{\cdots}{\rightarrow} (01)^n0$

$L(G) = \{x \in \{0,1\} / x = (01)^n0$ tel que $n \geqslant 1\}$

3) Une grammaire non ambigue

$S \rightarrow 01S \mid 0$ (Récursivité à droite)

$S \rightarrow 0101\underline{S} \rightarrow 010101S \rightarrow \cdots \rightarrow (01)^nS \rightarrow (01)^n0.$

$S \rightarrow 01S$ (Récursivité à gauche)

Exo4 :

G1 : $S \to bBAa / bB / aB$

$A \to aAbB$

$B \to SBaS / aC$

$C \to SBA / ab$.

(1) Elimination des symboles improductifs

$C \to ab \Rightarrow \{C\}$ productif

$B \to aC \Rightarrow \{B\}$ "

$S \to aB \Rightarrow \{S\}$ "

$\Rightarrow$ éliminer A.

(2) Elimination des symboles inaccessibles

$\{S\}$ par défaut

$\{S, B\}$ car $S \to aB$

$\{S, B, C\}$ car $B \to aC$

$$G' : \begin{cases} S \to bB / aB \\ B \to SBaS / aC \\ C \to ab \end{cases}$$

$$G2 : \begin{cases} S \to aSa \\ B \to Sb \mid bCC / DaB \\ C \to ab / DD \\ D \to aDB \\ E \to aC \end{cases}$$

(1) des symboles improductifs :

$C \to ab \Rightarrow$ c est productif. Suite de l'éxo 4

$B \to bCC \Rightarrow B$ " "

$E \to aC \Rightarrow E$ " "

$S \to aBa \Rightarrow S$ " "

D à éliminer

② Les symboles inaccessibles

$\{S\}$ par défaut

$\{S, B\}$ car $S \to aBa$

$\{S, B, C\}$ car $B \to bCC$

$\{S, B, C, E\}$ car $E \to aC$

$$G_2 : \begin{cases} S \to aBa \\ B \to Sb / bCC \\ C \to ab \\ E \to aC \end{cases}$$

Exo5 :

Les grammaires régulières qui génèrent les langages suivants

$L1 = (01)^*$

$G1 = \{ \{S\}, \{0,1\}, P, S \}$ où

$$P \begin{cases} S \longrightarrow 01S \\ S \longrightarrow \varepsilon \end{cases}$$

$L2 = (01)^* 11$

$G2 = (\{S\}, \{0,1\}, P, S)$ ou $P: S \rightarrow 01S / 11$.

$L3 = 0^* + (0+11)^*$

$G3 = (\{S, S1, S2\}, \{0,1\}, P, S\}$ où

$$P: \begin{cases} S \rightarrow S1 / S2 \\ S1 \rightarrow 0S1 / \varepsilon \\ S2 \rightarrow 0S2 / 11S2 / \varepsilon \end{cases}$$

$L4 = 0^* 1^*$

$G4 = \{ \{S, S1, S2\}, \{0,1\}, P, S\}$ où

$$P: \begin{cases} S \rightarrow S_1 S_2 \\ S_1 \rightarrow 0S_1 \backslash \varepsilon \\ S_2 \rightarrow 1S_2 \backslash \varepsilon \end{cases}$$

Suite exo6 :

* $L_6 = \{0^n 1^n 0^m / n \geq 0 ; m \geq 0\}$

$$\begin{cases} S \rightarrow S_1 S_2 \\ S_1 \rightarrow 0 S_1 1 / \varepsilon \\ S_2 \rightarrow 0 S_2 / \varepsilon \end{cases}$$

Exo7 :

$G = (\{S, A, B\}, \{a, b\}, P, S)$

$$\begin{cases} S \to bA \mid aB \\ A \to bAA \mid aS \mid a \\ B \to aBB \mid bS \mid b \end{cases}$$

1) aabbab $\overset{?}{\in} L(G)$

$S \to a\underline{\underline{B}} \to aa\underline{\underline{B}}B \to aab\underline{\underline{S}}B \to aabb\underline{\underline{A}}B \to aabbaB$
$\to aabbab$ est accepté par L.

2) Ce langage génère autant de a que de b dans un ordre quelconque

3) Forme Normale de Chomsky.

G(V, T, P, S) grammaire hors contexte.
On dit G est sous G.N.C si toutes les productions de P sont de la forme :

$$\begin{cases} A \to BC & \text{où } A, B, C \in V \text{ et } a \in T \\ A \to a \end{cases}$$

$S \to bA \mid aB$
$A \to bAA \mid aS \mid a$
$B \to aBB \mid bS \mid b$

$\Rightarrow$ F.N.C

$$\begin{cases} S \to Y_b A \mid Y_a B \\ A \to Y_b C \mid Y_a S \mid Y_a \\ C \to AA \\ B \to Y_a D \mid Y_b S \mid Y_b \\ D \to BB \\ Y_a \to a \\ Y_b \to b \end{cases}$$

Exo 8 :

$A \rightarrow Aab \mid Aba \mid a \mid b$

(1) Pour éliminer la récursivité à gauche ; on applique

$$\begin{cases} A \rightarrow A\alpha \\ A \rightarrow \beta \end{cases} \equiv \begin{cases} A \rightarrow \beta A' \mid \beta \\ A' \rightarrow \alpha A' \mid \alpha \end{cases}$$

$$A \rightarrow Aab \mid Aba \mid a \mid b$$

$$\Updownarrow$$

$$A \rightarrow A\alpha_1 \mid A\alpha_2 \mid \beta_1 \mid \beta_2$$

$$\begin{cases} A \rightarrow \beta_1 A' \mid \beta_2 A' \mid \beta_1 \mid \beta_2 \\ A' \rightarrow \alpha_1 A' \mid \alpha_2 A' \mid \alpha_1 \mid \alpha_2 \end{cases}$$

$$\begin{cases} A \rightarrow aA' \mid bA' \mid a \mid b \\ A' \rightarrow baA' \mid abA' \mid ba \mid ab \end{cases}$$

Cette grammaire non récursive à gauche.

(2) $A \rightarrow Aab \mid Aba \mid a \mid b$

Le langage généré par cette grammaire est

$L = (a+b)(ab+ba)^*$

Exo 9 :

Soit $G = (\{S, A\}, \{a, b\}, P, S)$

$$\begin{cases} S \rightarrow AA \mid a \\ A \rightarrow SS \mid b \end{cases}$$

G est sous forme normale de Greibach si :

* G est hors contexte
* toutes les productions P de la forme

$A \rightarrow a\alpha$ tel que $A, \alpha \in V^*$.

$a \in T$.

Suite de l'exo 9

$$\begin{cases} S \to AA \mid a \\ A \to SS \mid b \end{cases}$$

$A_1 = S$; $A_2 = A$

$$\begin{cases} A_1 \to A_2A_2 \mid a \\ A_2 \to A_1A_1 \mid b \end{cases}$$

On remplace A_1 par $A_1 \to A_2A_2$ et $A_1 \to a$.

$A_2 \to A_2A_2A_1 \mid b \mid aA_1$

$A_2 \to A_2 \underbrace{A_2A_1}_{\alpha} \mid \underbrace{b}_{\beta_1} \mid \underbrace{aA_1}_{\beta_2}$

$A_2 \to A_2\alpha \mid \beta_1 \mid \beta_2$.

On élimine la recursivité à gauche.

$$\begin{cases} A_2 \to \beta_1A_3 \mid \beta_2A_3 \mid \beta_1 \mid \beta_2 \\ A_3 \to \alpha A_3 \mid \alpha \end{cases}$$

$$\begin{cases} A_2 \to aA_1A_3 \mid bA_3 \mid aA_1 \mid b \\ A_3 \to A_2A_1A_3 \mid A_2A_1 \\ A_1 \to A_2A_2 \mid a \end{cases}$$

$$\begin{cases} A \to aSA_3 \mid bA_3 \mid aS \mid b \\ A_3 \to ASA_3 \mid AS \\ S \to AA \mid a \end{cases}$$

$$\begin{cases} A \to aSA_3 \mid bA_3 \mid aS \mid b \\ A_3 \to aSA_3SA_3 \mid bA_3SA_3 \mid aSSA_3 \mid bSA_3 \mid aSA_3S \mid aSS \mid bA_3S \mid b \\ S \to aSA_3A \mid bA_3A \mid aSA \mid bA \mid a \end{cases}$$

EXO 10 :

$G = (\{S, B, C\}, \{a, b, c\}, P, S)$

$S \to aSBC \mid abC$ (R1) (R'1)

$CB \to BC$ (R2)

$bB \to bb$ (R3)

$bC \to bc$ (R4)

$cC \to cc$ (R5)

1) Cette grammaire est de type 1 « sensible au contexte »

2) $a^2b^2c^2$ et $a^3b^3c^3$ sont générés par G

$S \xrightarrow{R_1} aSBC \xrightarrow{R'_1} aabCBC \xrightarrow{R_2} aabBCC \xrightarrow{R_3} aabbCC \xrightarrow{R_4} aabbcC$

$\xrightarrow{R_5} aabbcc$

$S \xrightarrow{R_1} aSBC \xrightarrow{R_1} aaSBCBC \xrightarrow{R'_1} aaabCBCBC \xrightarrow{R_2} aaabBCCBC$

$\xrightarrow{R_3} aaabbCCBC \xrightarrow{R_2} aaabbCBCC \xrightarrow{R_2} aaabbBCCC \xrightarrow{R_3}$

$aaabbbCCC \xrightarrow{R_4} aaabbbcCC \xrightarrow{R_5} aaabbbccC \xrightarrow{R_5} aaabbbccc$

3) Le langage généré par cette grammaire est :

$L(G) = \{a^n b^n c^n / n \geq 2\}$.

Exo 11 :

$L_1 = \{ wcw^t / w \in (0+1)^* ; x \text{ et } y \in (0+1) \}$.

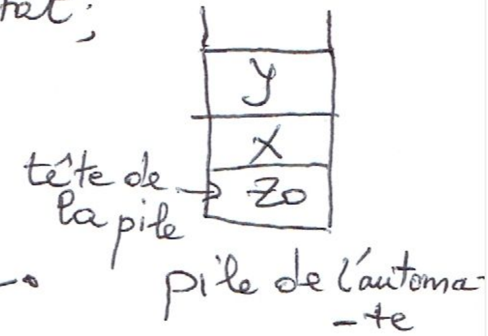
$\delta(q_0, x, Z_0) = (q_0, xZ_0)$; empiler x ;

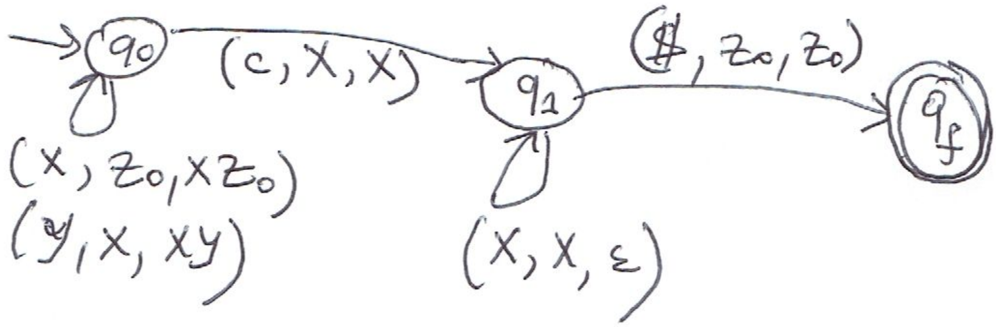
$\delta(q_0, y, x) = (q_0, yx)$; On empile le reste ;

$\delta(q_0, c, x) = (q_1, x)$; On change l'etat ;

$\delta(q_1, x, x) = (q_1, \varepsilon)$; On dépile.

$\delta(q_1, \$, Z_0) = (q_f, Z_0)$; On accepte.





$L_2 = \{ 0^n 1^{2n} / n \geq 0 \}$

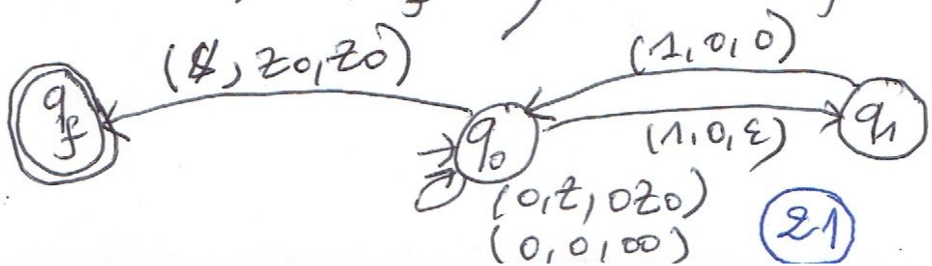
$\delta(q_0, 0, Z_0) = (q_0, 0Z_0)$; On empile le 1er zero.

$\delta(q_0, 0, 0) = (q_0, 00)$; On empile le reste des zéros.

$\delta(q_1, 1, 0) = (q_1, \varepsilon)$; On dépile.

$\delta(q_1, 1, 0) = (q_1, 0)$; On fait Rien.

$\delta(q_1, \$, Z_0) = (q_f, Z_0)$; On accepte.



$L_6 = \{ 0^n 1^n 0^m \,/\, n \geqslant 0 \,;\, m \geqslant 0 \}$.

$\delta(q_0, 0, Z_0) = (q_0, 0Z_0)$; Empiler le 1er zéro.

$\delta(q_0, 0, 0) = (q_0, 00)$; Empiler le reste des zéros.

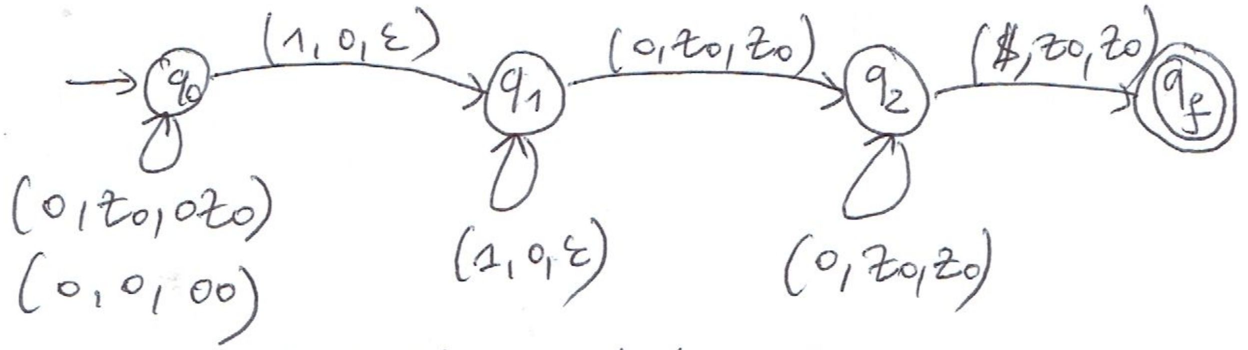
$\delta(q_0, 1, 0) = (q_1, \varepsilon)$; On dépile 0

$\delta(q_1, 1, 0) = (q_1, \varepsilon)$; On dépile tous les zéros

$\delta(q_1, 0, Z_0) = (q_2, Z_0)$; On change l'état

$\delta(q_2, 0, Z_0) = (q_2, Z_0)$; Ne rien faire

$\delta(q_2, \$, Z_0) = (q_f, Z_0)$; On accepte.



② $L = \{ |w_a| = |w_b| \,/\, w \in (a+b)^* \}$.

$\delta(q_0, a, Z_0) = (q_0, aZ_0)$; Empiler le 1er a ;

$\delta(q_0, b, Z_0) = (q_0, bZ_0)$; Empiler le 1er b ;

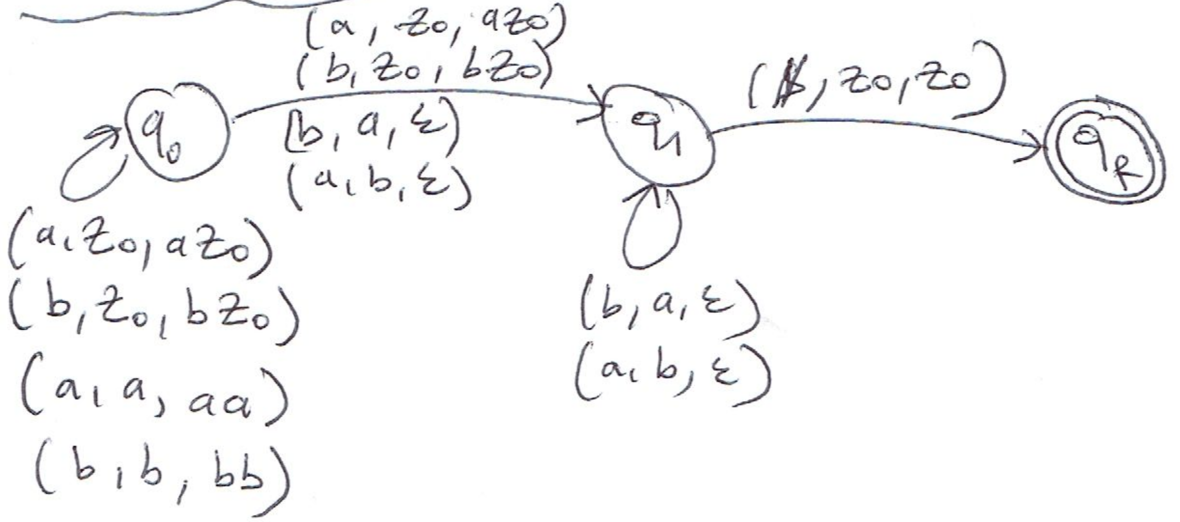
$\delta(q_0, a, a) = (q_0, aa)$; On empile les a ;

$\delta(q_0, b, a) = (q_1, \varepsilon)$; On dépile ;

$\delta(q_1, b, a) = (q_1, \varepsilon)$; On dépile tous les a ;

$\delta(q_1, \$, Z_0) = (q_f, Z_0)$; On accepte.

Suite de l'exo 11



La suite des configurations de l'automate pour le mot aabbabab.

$(q_0, aabbabab\$, Z_0) \vdash (q_0, abbabab\$, aZ_0)$

$\vdash (q_0, bbabab\$, aaZ_0) \vdash (q_1, babab, aZ_0)$

$\vdash (q_1, abab\$, Z_0) \vdash (q_0, bab\$, aZ_0) \vdash (q_1, ab\$, Z_0)$

$\vdash (q_0, b\$, aZ_0) \vdash (q_1, \$, Z_0).$