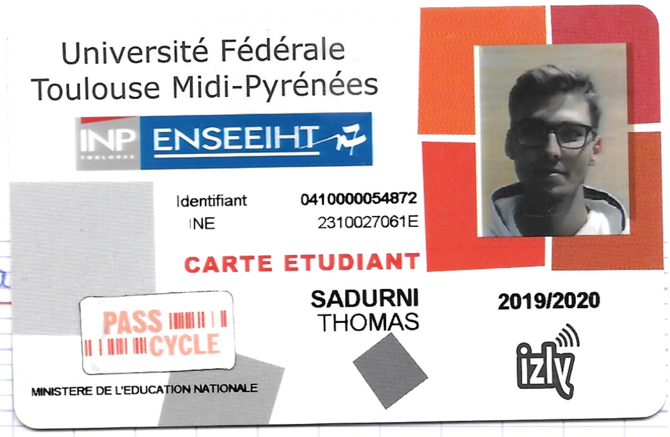
Thomas
SADURNI
2SN[illegible]

Examen Automa



Exercice 1

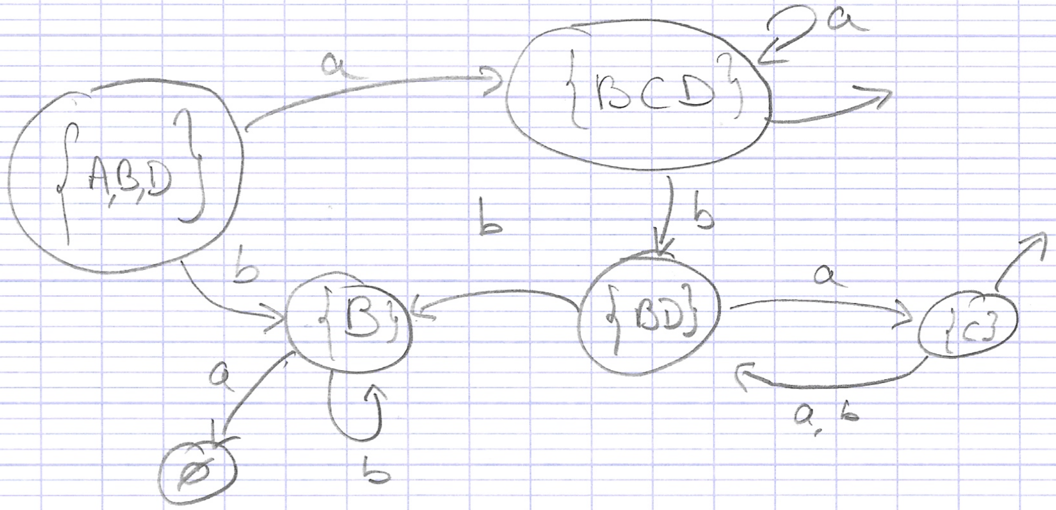
1. $\mathcal{E}$ n'est pas déterministe car on peut apercevoir deux états initiaux ou des epsilon - transitions.

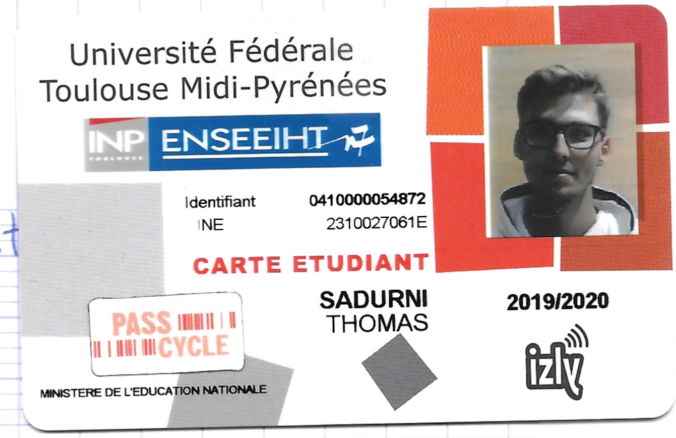
Déterminisons le :
Etablissons d'abord sa table de transition

	$\varepsilon F()$	a	b
A	A	B, C, D	$\emptyset$
B	B, D	$\emptyset$	B
C	C	D, B	B, D
D	D, B	C	$\emptyset$

$$\varepsilon F(\{A, D\}) = \{A, D, B\} \quad (\text{état initial})$$

On peut maintenant construire l'automate





2 La table de transition est

	a	b
{ABD}	{BCD}	{B}
{BCD}	{BCD}	{BD}
{BD}	{C}	{B}
{B}	{B}	∅
{C}	{BD}	{BD}

Exercice 2

1. Le système d'équation est :

$$\begin{cases} E = aG + bE \\ F = aE + bF \\ G = bF + \Lambda \end{cases}$$

2.
$$\begin{aligned} F &= a(aG + bE) + bF \\ &= a(a(bF+\Lambda) + bE) + bF \\ &= aabF + aa + abE + bF \\ &= (aab+b)F + aa + abE \\ &= (aab+b)^*(aa + abE) \end{aligned}$$

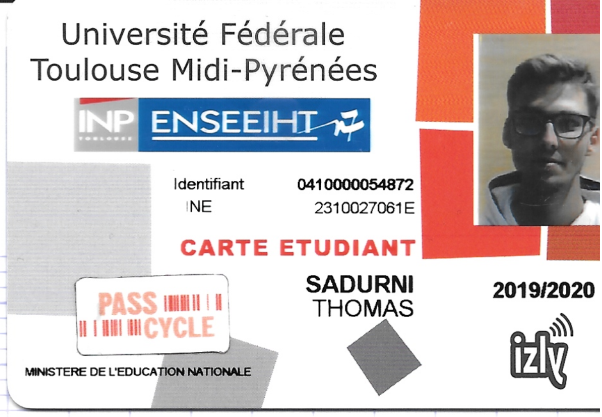
$$\Rightarrow G = b(aab+b)^*(aa+abE) + \Lambda$$

$$\begin{aligned} \Rightarrow E &= a\left(b(aab+b)^*(aa+abE)+\Lambda\right) + bE \\ &= \left(ab(aab+b)^*(aa+abE)+a\right) + bE \\ &= ab(aab+b)^* aa + abE + a + bE \\ &= ab(aab+b)^* aa + a + (ab+b)E \end{aligned}$$

$$E = (ab+b)^*\left(ab(aab+b)^* aa + a\right)$$

Thomas SADURNI

Exercice 3



$\mathcal{L} = (a|\Lambda)((ba)*|ba)$

Soit $X_0 = (a|\Lambda)((ba)*|ba)$

- $D_a(X_0) = (ba)*|ba = X_1$

  $D_b(X_0) = \emptyset \mid D_b(ba)(ba)* \mid a$

  $D_b(X_0) = a(ba)*|a = X_2$

- $D_a(X_1) = \emptyset$

  $D_b(X_1) = X_2$

- $D_a(X_2) = D_a(a(ba)*|a) = D_a(a(ba)*)|D_a(a)$

  $= (ba)*|\Lambda = X_3$

  $D_b(X_2) = \emptyset$

- $D_a(X_3) = \emptyset$

  $D_b(X_3) = D_b(ba)(ba)* \mid \emptyset$

  $= a(ba)* = X_4$

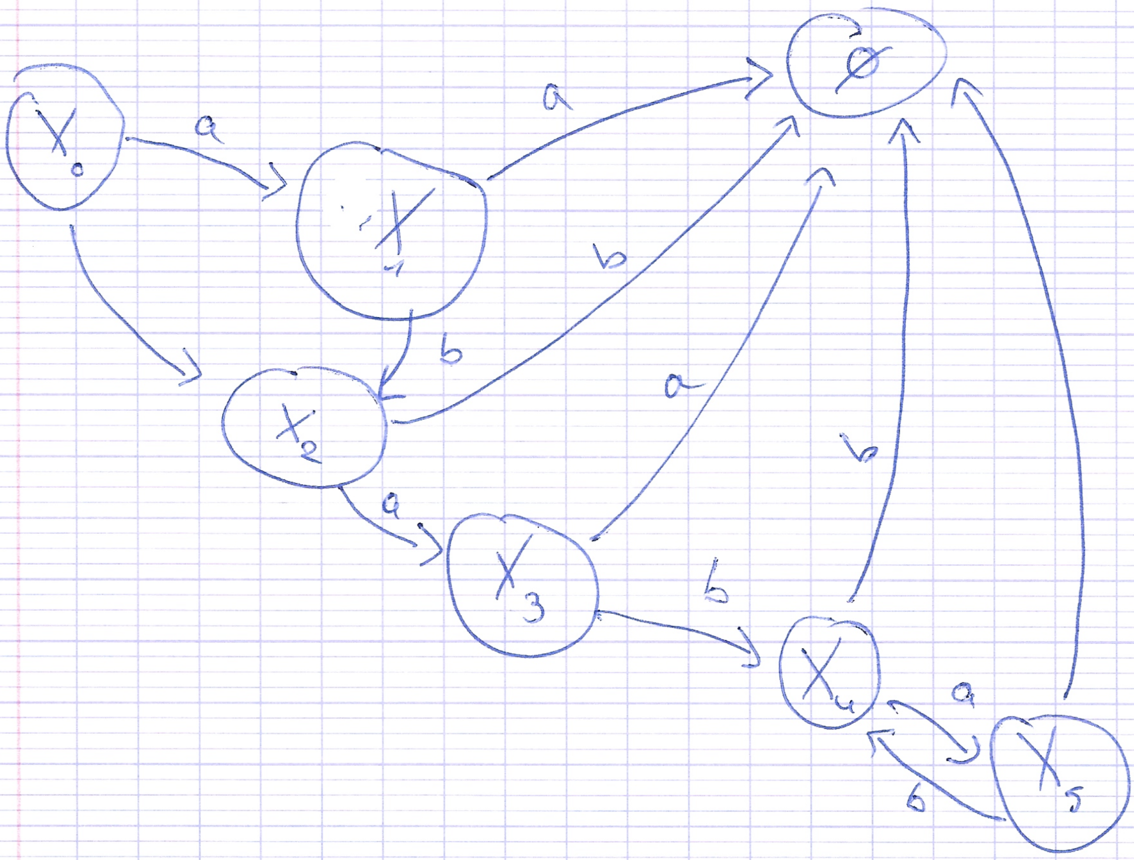
- $D_a(X_4) = (ba)* = X_5$

  $D_b(X_4) = \emptyset$

- $D_a(X_5) = \emptyset$

  $D_b(X_5) = a(ba)* = X_4$

L'automate représentant ce langage est :



Exercice 4

$$\begin{cases} N \to R\,p\,I \\ N \to R \\ N \to I \\ I \to R\,i \\ R \to R\,c \\ R \to c \end{cases} \Rightarrow \begin{cases} N \to R\,p\,I \mid R \mid I \\ I \to R\,i \\ R \to R\,c \mid c \end{cases}$$

$$\Rightarrow \begin{cases} N \to R\,p\,I \mid R \mid I \\ I \to R\,i \\ R \to c\,R' \\ R' \to c\,R' \\ R' \to \Lambda \end{cases}$$

$N \to R\,p\,I \mid R \mid I$

$\quad\to cR'\,pI \mid cR' \mid cR'i$

$\quad\to c\,(R'\,p\,I \mid R' \mid R'i)$

$N \to cF$

$F \to R'\,pI \mid R' \mid R'i$

$\quad\to cR'\,pI \mid pI \mid cR' \mid cR'i \mid i \mid \Lambda$

$\quad\to c\,(R'\,pI \mid R' \mid R'i) \mid pI \mid i \mid \Lambda$

$F \to cF \mid pI \mid iI \mid \Lambda$

$I \to Ri$ avec $G \to cR'i \mid i\Lambda$

$\quad\to cR'i \qquad\qquad \to cG \mid i\Lambda$

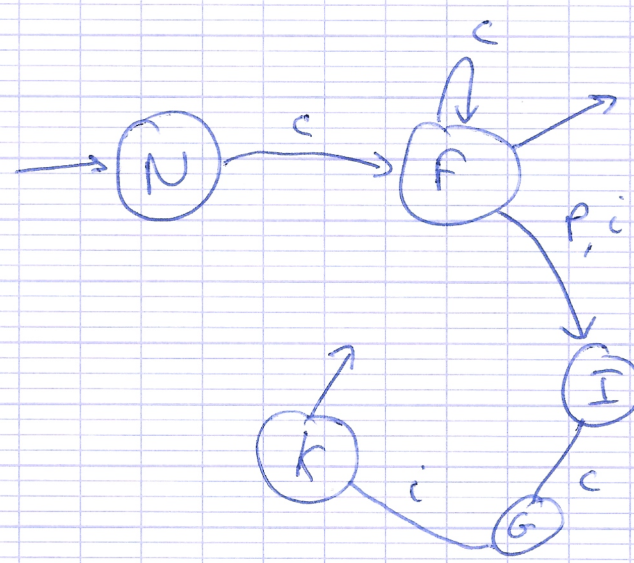
$\quad\to cG \qquad\qquad \to cG \mid iK$

avec $K \to \Lambda$

G_m a donc :

$$\begin{cases} N \to cF \\ F \to cF \\ F \to pI \\ F \to iI \\ F \to \Lambda \\ I \to cG \\ G \to cG \\ G \to iK \\ K \to \Lambda \end{cases}$$

L'automate est le suivant :



Exercice 5

1 a/b La grammaire est recursive à gauche

$$\begin{cases} O \to \{\} \\ O \to \{L\} \\ L \to L , F \\ L \to F \\ F \to id : num \end{cases} \Rightarrow \begin{cases} O \to \{\} \\ O \to \{L\} \\ L \to L , F \mid F \\ F \to id : num \end{cases}$$

$G_1 \Rightarrow$

$$\begin{cases} O \to \{\} \\ O \to \{L\} \\ L \to F L' \\ L' \to , F L' \\ L' \to \Lambda \\ F \to id : num \end{cases} \qquad \begin{array}{l} SD(O \to \{\}) = \\ SD(O \to \{L\}) = \text{"\{"} \\ SD(L \to F L') = \text{"id"} \\ SD(L' \to , F L') = \text{","} \\ SD(L' \to \Lambda) = S(L') = \text{"\}"} \\ SD(F \to id : num) = \text{"id"} \end{array}$$

c G_1 est $LL(1)$ car l'ensemble des regles definissant un meme symbole non terminal ont des symboles directeurs disjoints

2a

b) L'analyseur lexicale delivre à l'analyseur syntaxique un flot d'entrée sous forme de caractères individuels

c) L'analyseur syntaxique verifie que le programme est conforme à la grammaire