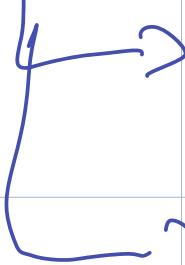


Hier die de Chancry

HOM

primaires  
automates

formelles



Correcteur orthographique

R I

GN  
GV

cyber - physique

Machine à sole

mécanique  
informique

algorithmes  
circuits

avion / voiture

algorithmes  
+ langage naturel.



Type O généraux

apparence

type 1 contextuel

Type 2  
hors contexte  
non contextuels

type 3  
regulier  
infinielle

Le langage de type i

ssi

lexique une grammaire de type i  
qui produit ce langage

$$G_1 = S \rightarrow a \quad | \quad S \quad | \quad aS$$

Type 2

$$G_2 \quad S \rightarrow a \quad | \quad aS$$

Type 3 si elle engendre le même langage

$$L(G_1) = L(G_2) = a^+$$

$a^+$  est de type 3 ( regulier )

expr régulières : écrit des langages réguliers

terminaux

!

( )<sup>d</sup>

union  
concaténation

\* de Kleene

$(ab + c)^*$  ba  
abs c ab ab ba

automate d'états finis  
à -

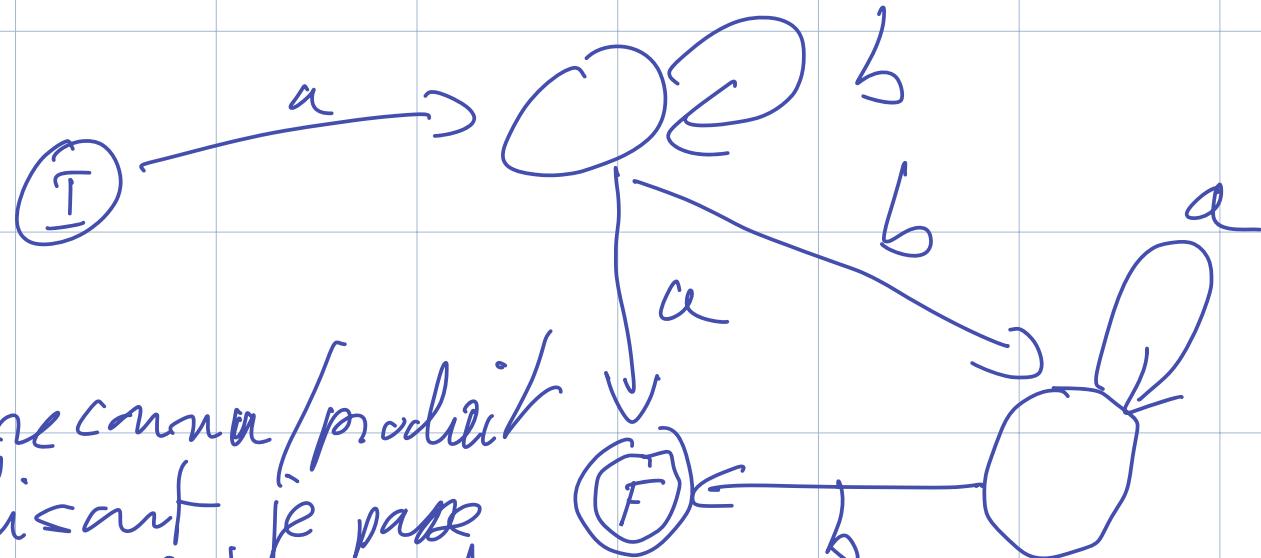
automate fini       $a b^* (ba^+b) a$

un état initial

un ou des état final aux acceptation

transitions dirigées par l'alphabet

$\Sigma = \{a, b\}$

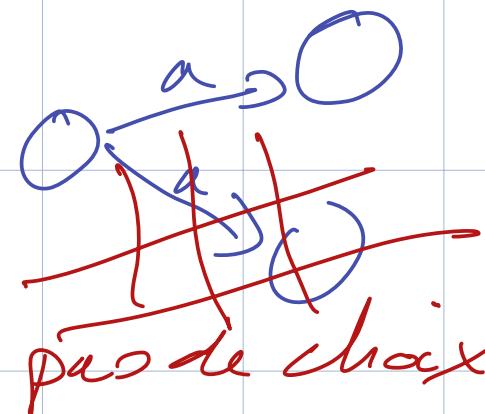
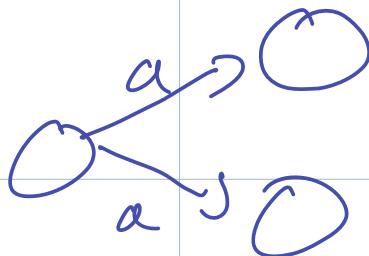


un mot est reconnu/produit  
si on le lisant je passe  
de I à un état final

Variantes Equivalentes  
(decrivent les mêmes langages)

- un ou plusieurs état final

• non déterministe  $\rightarrow$  déterministe  
n états  $\rightarrow$   $2^n$  états



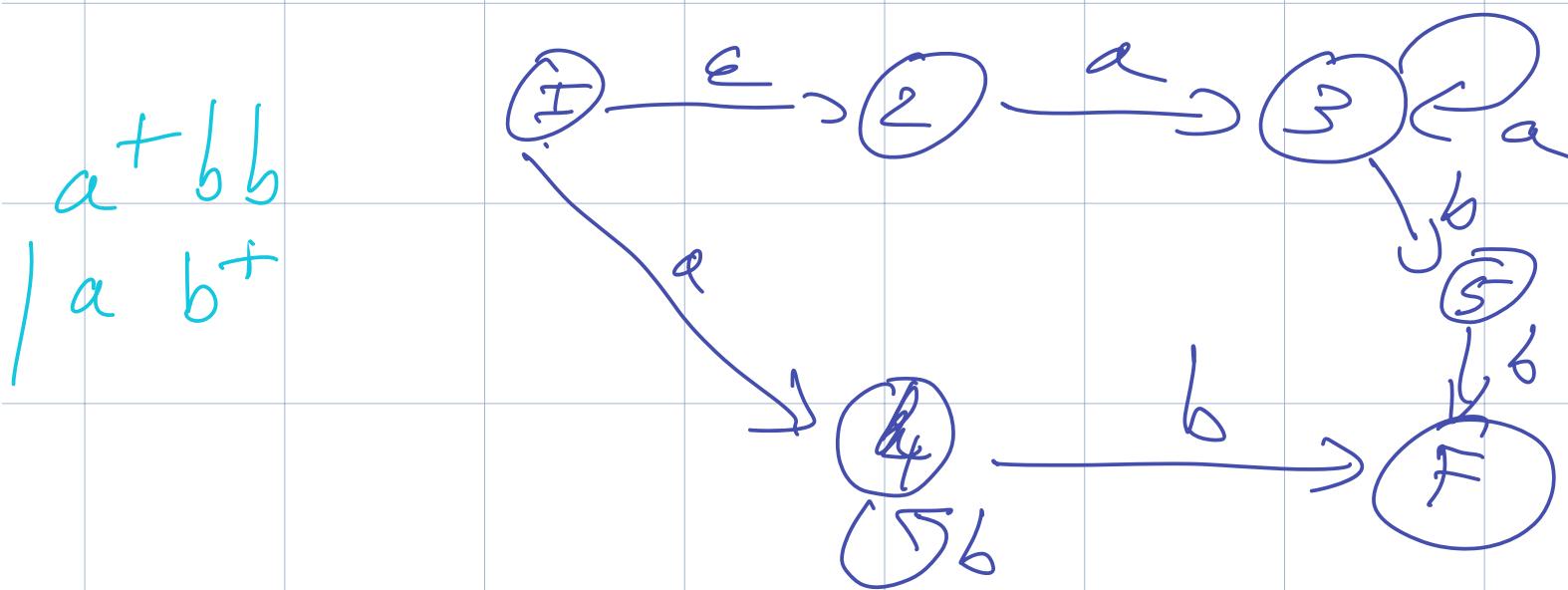
• il existe nécessairement une transition entre deux états consécutifs

• avec  $\epsilon$  transition  $\rightarrow$  sans  $\epsilon$  transition

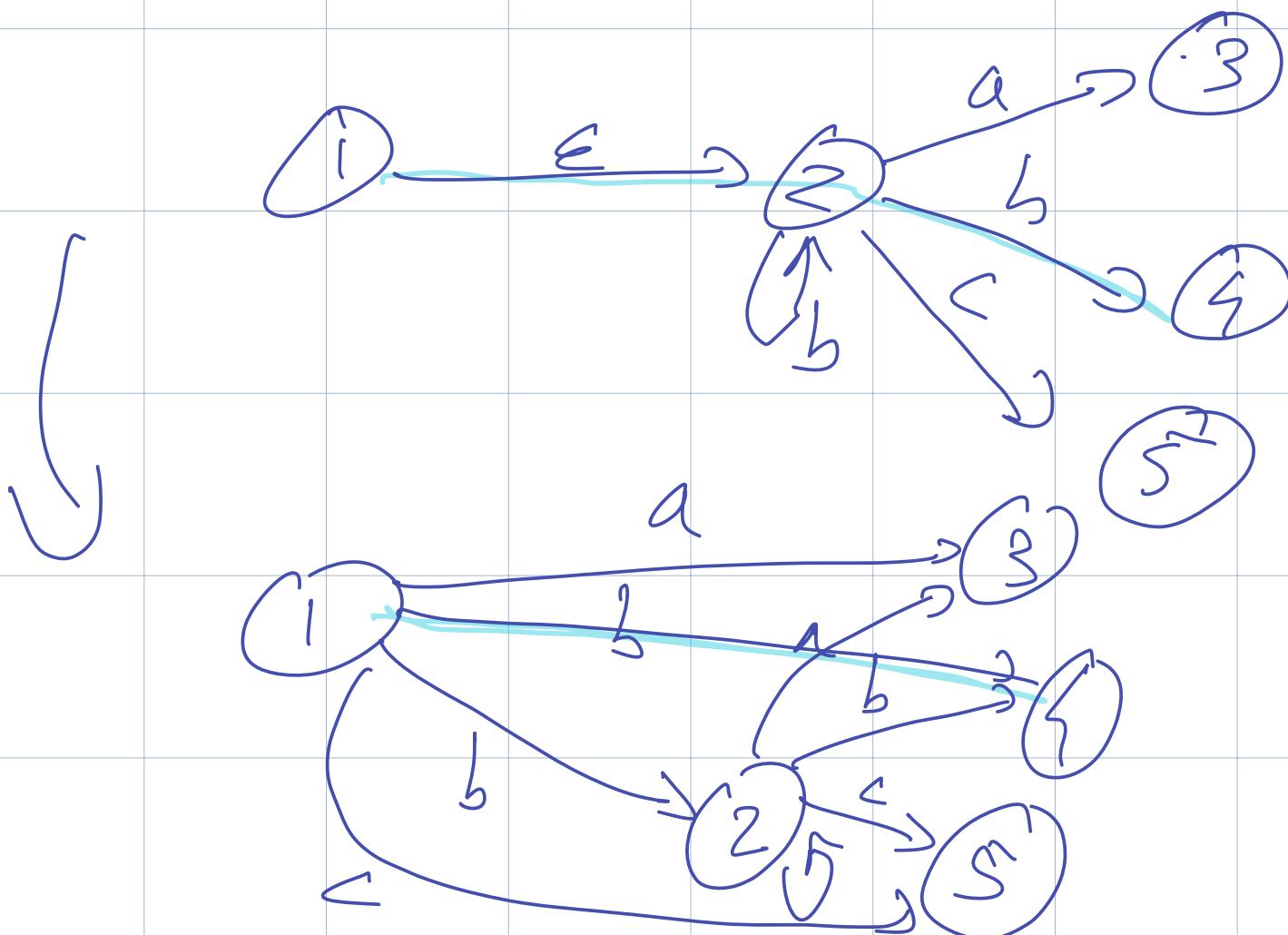
$\epsilon$  transition :



Pas avancer dans le mot  
on "saut" d'un état à un autre

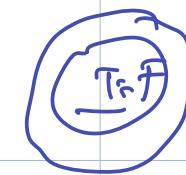
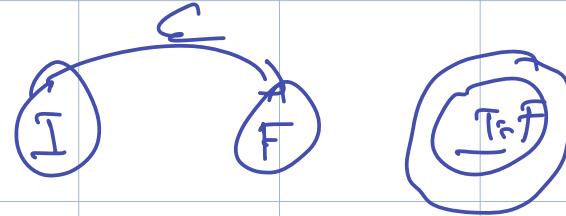


Un automate avec des  $\epsilon$ -transitions peut être converti en un automate sans  $\epsilon$  transition.

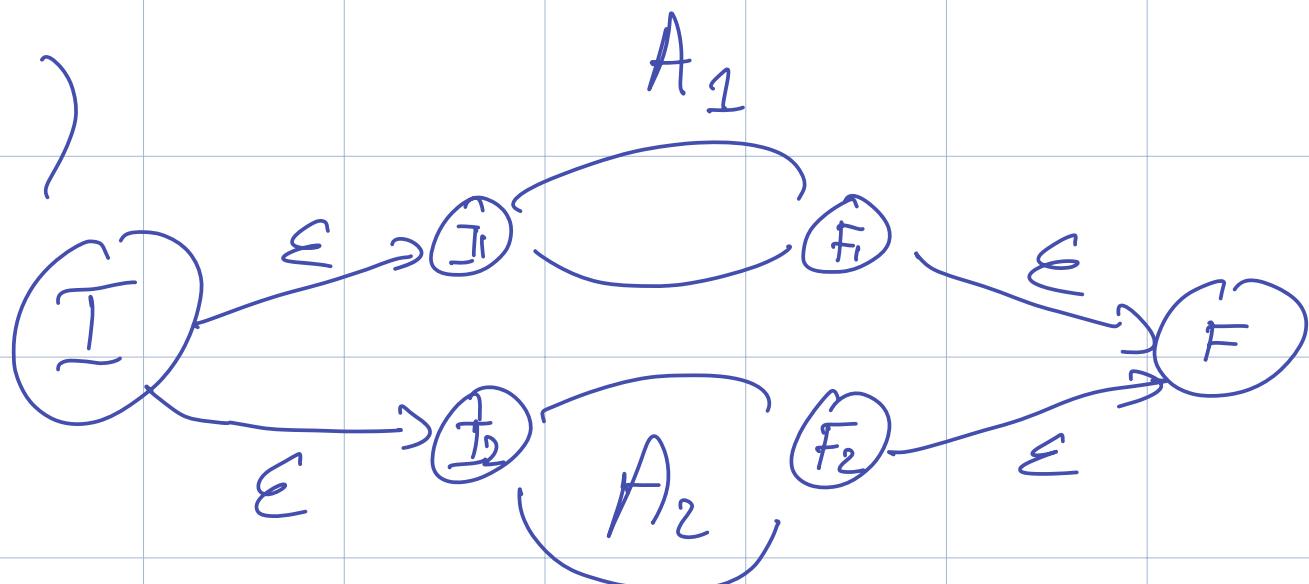


Expr régulière  $\rightarrow$  automate à 1 état final des E-transitions

Expr rég

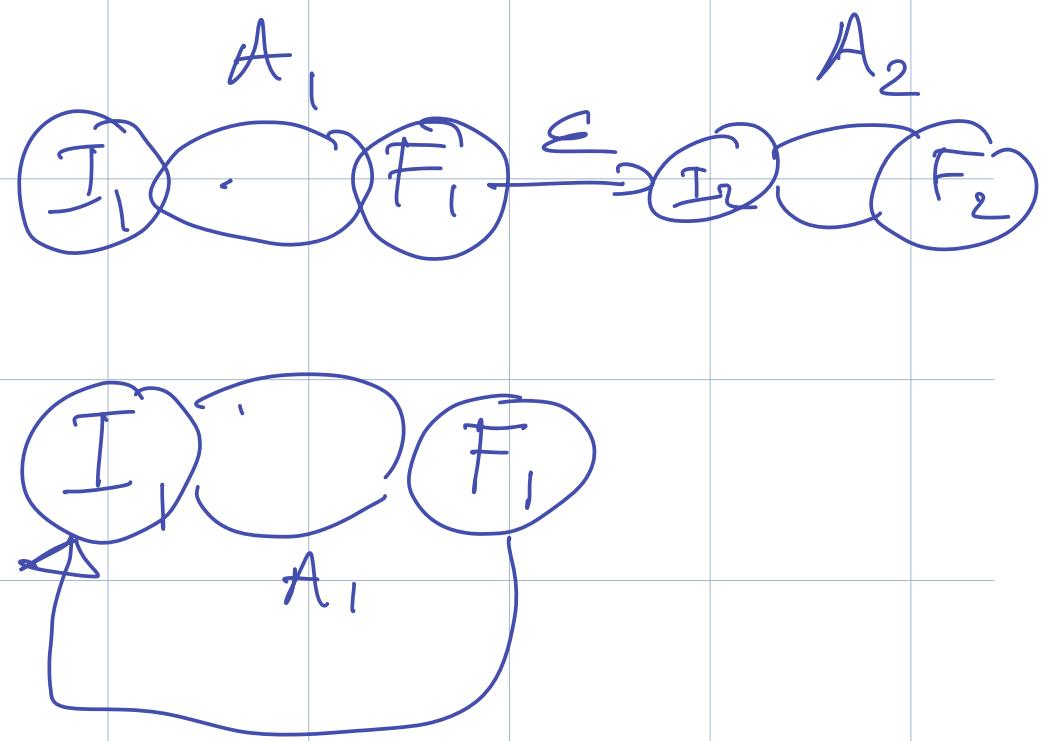


$(e_1 | e_2)$



$$e_1 \cdot e_2$$

$$(e_1)$$



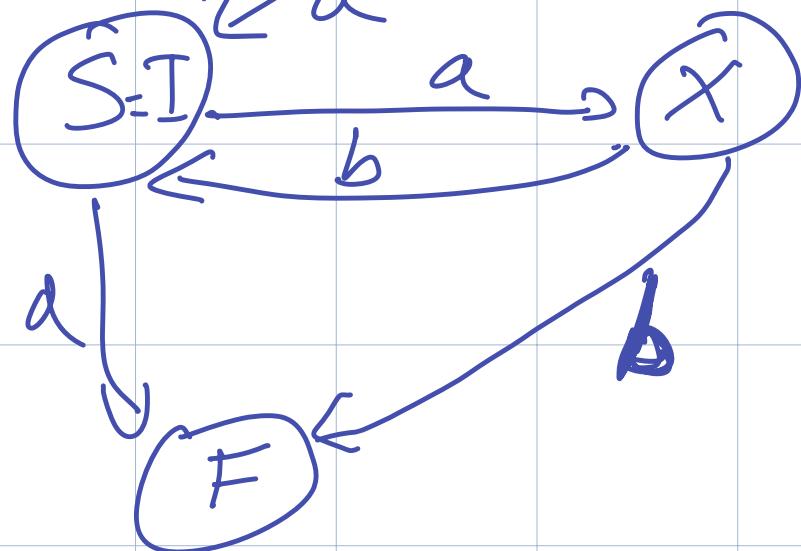
On a vu

Quelques

exp. régulières

autonome  $\leftrightarrow$  grammaire régulières

$$\begin{array}{l} S \rightarrow aS \quad | \quad aX \quad | \quad a \\ X \rightarrow b \quad | \quad bS \end{array}$$

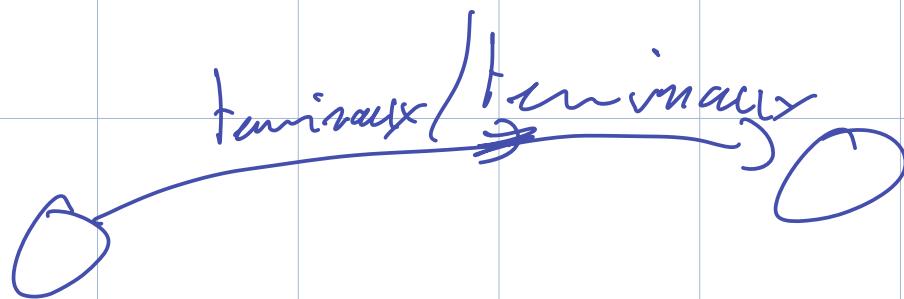


Autonomes

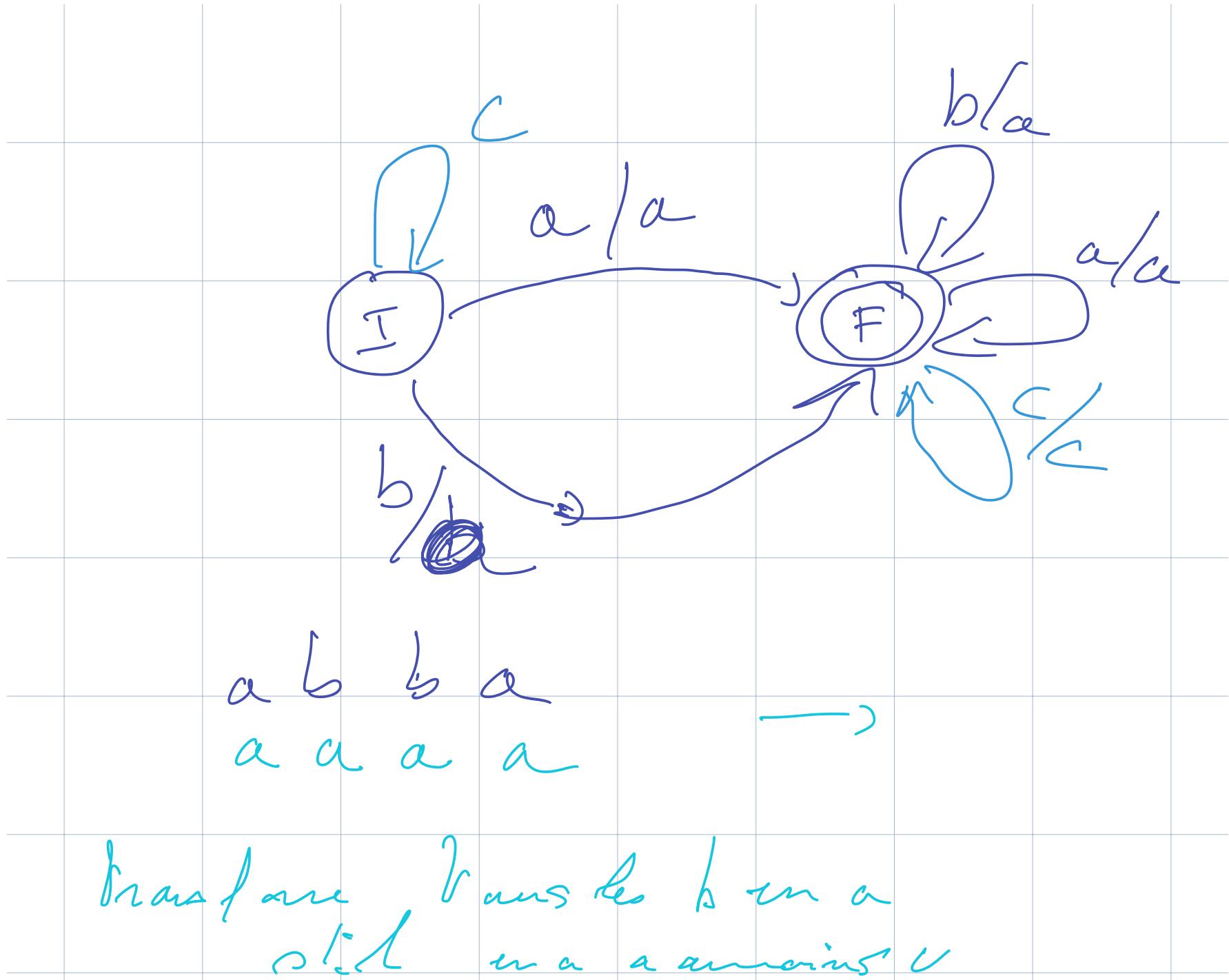
I/O

interprétable

transducteurs.



transforms le mot en



# Morphologie flexionnelle.

- conjugaison

- déclinaison

- plural / sing  
mascul / fem.

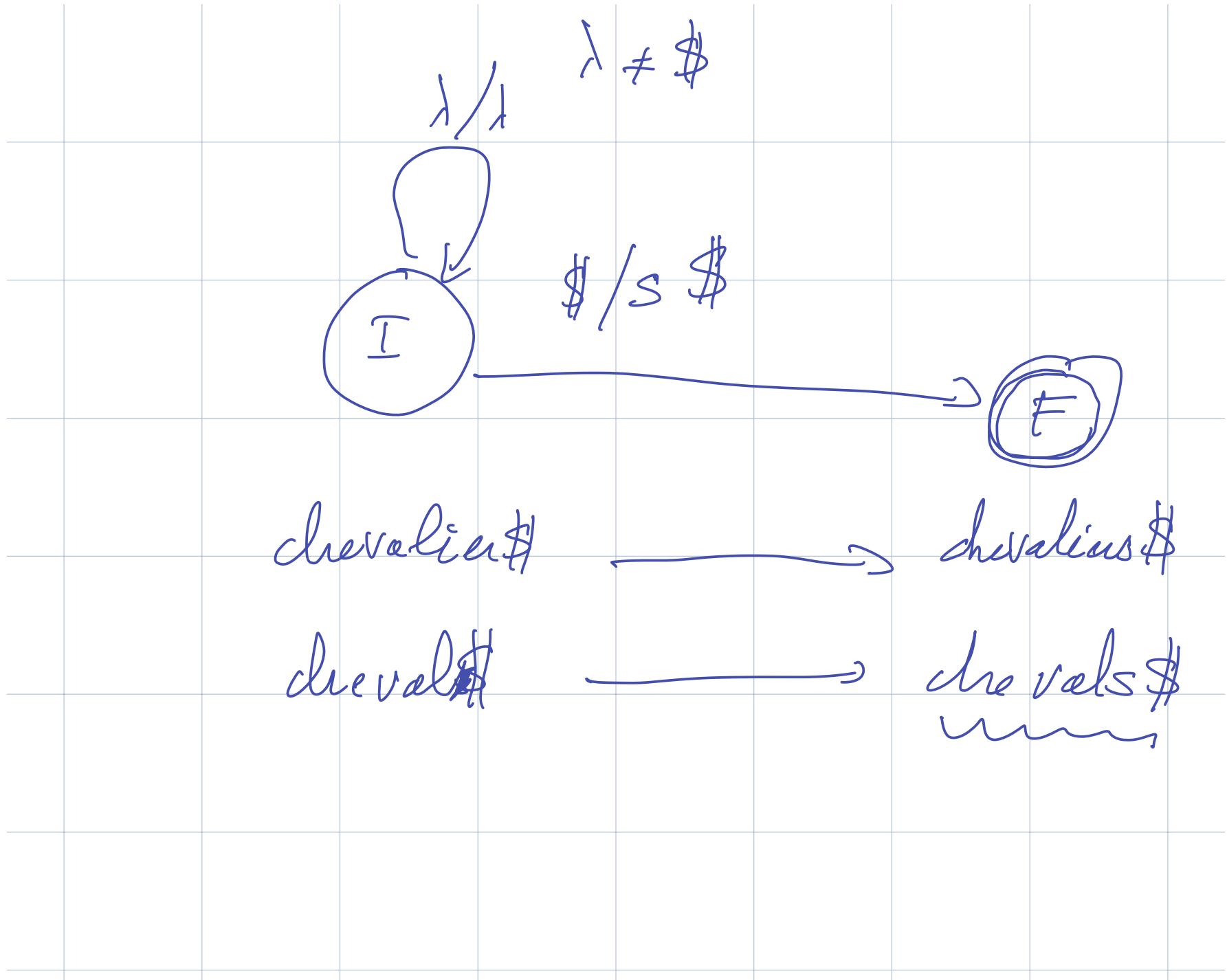
y

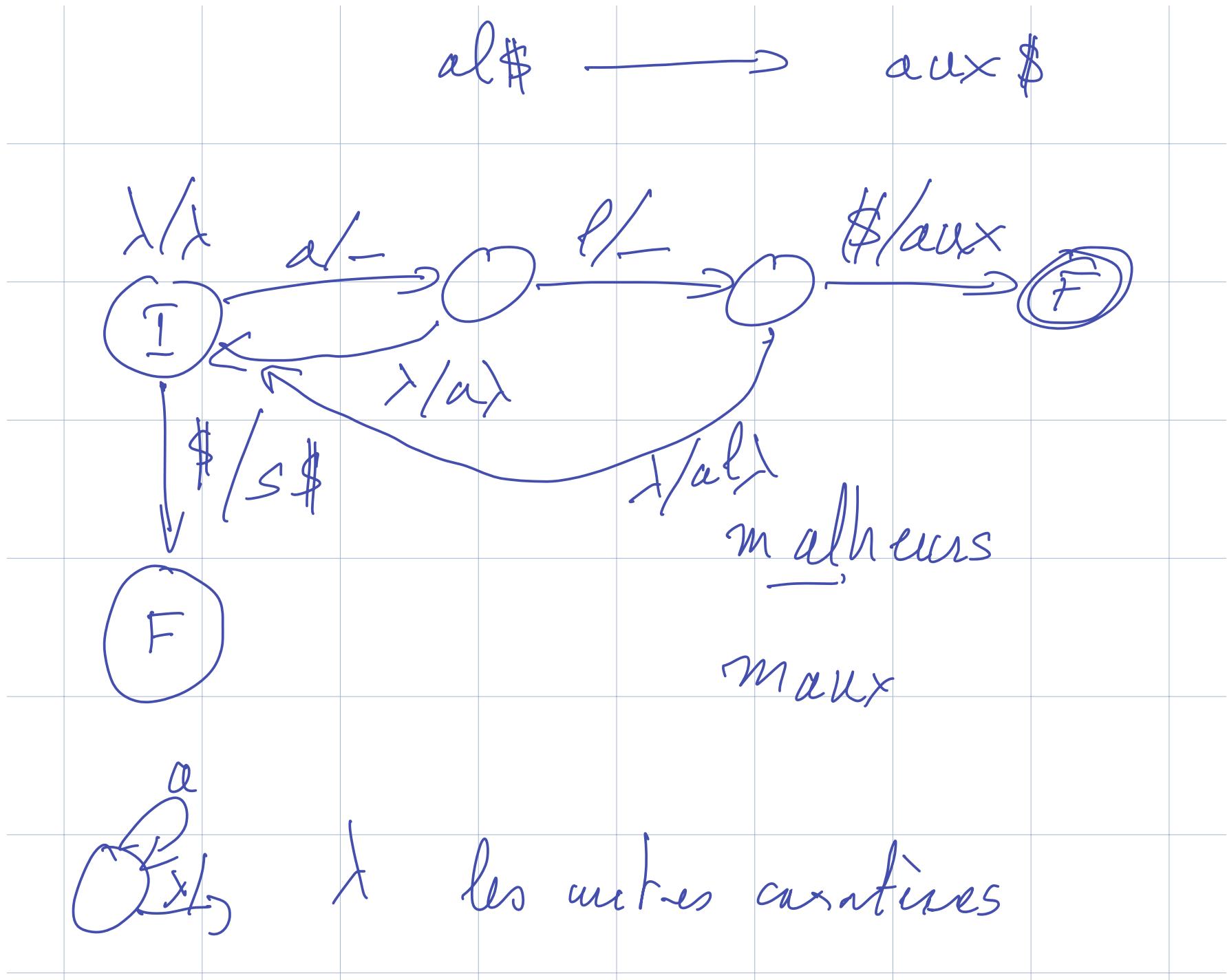
en

A utomatics qui met un nom  
au pluriel

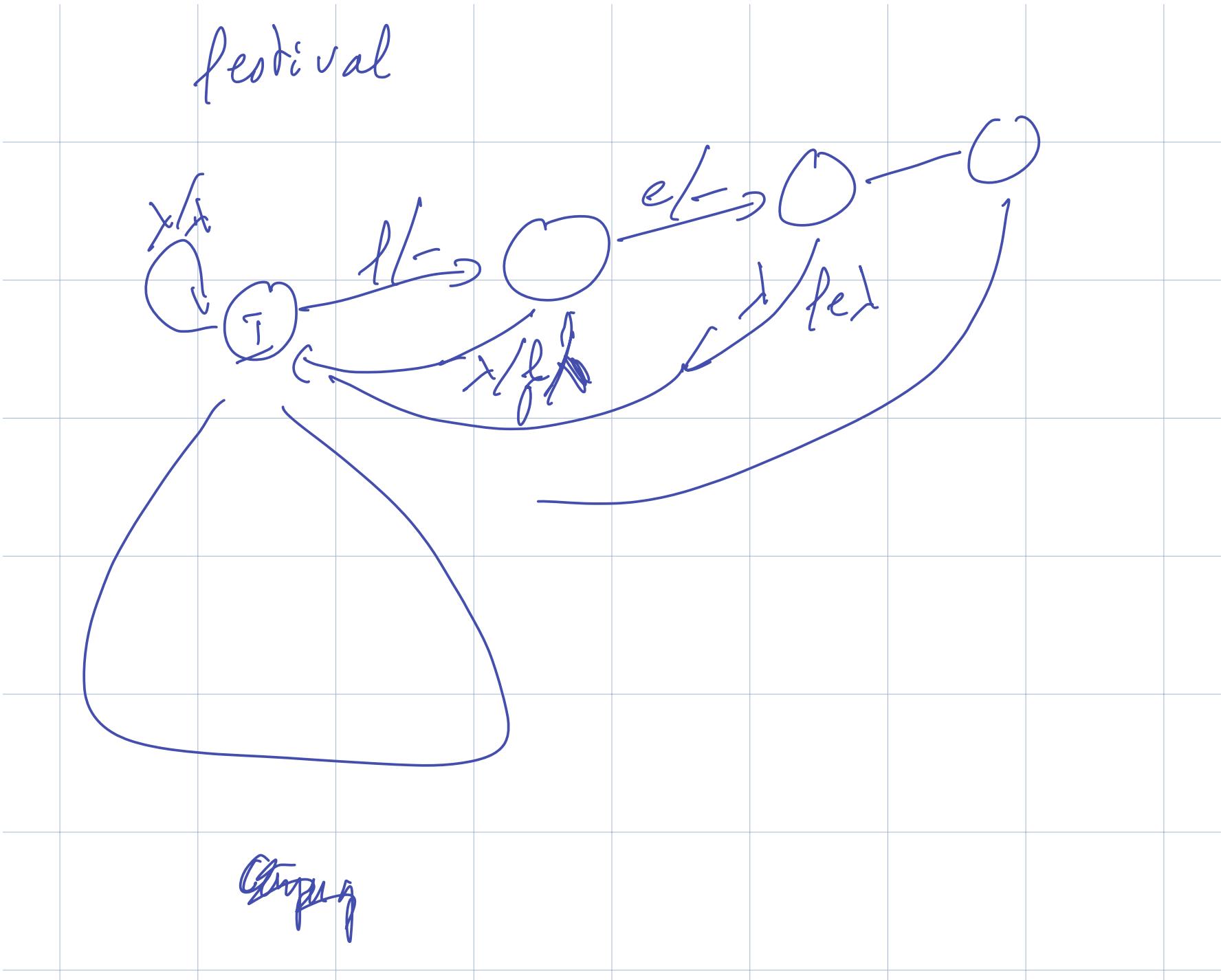
(on ajoute un s à la fin)

Signe de fin de mot \$





festival



Si elles sont des firmes

leur quelques un corpos

\* apprendre ok

sinon autoroutes

pluie de membres

quatre-vingt

trois cents

trois cent seize

(présent)      /      passe composé / futur  
sujet présent.      bcp en cours

coréction anthropomorphique

pas de machine learning

- trop de calcul
- trop d'erreurs dans le capteur

après qu'il soit arrivé, si qu'il  
est arrivé

RI

production de lait

production

laitier

production minière

production de minéraux

automates

dans  
Google  
6-15 ans

Synapse

motivé de  
recherche

intelligence

~~mines~~  
~~mineraux~~

grammaire

hors contexte

presque  
pour

suffisant

au syntaxe de la phrase

~~- polyvalent~~ - ~~Type 1~~  
~~contextuel~~

decidable

Type 2  
hors contexte

Languages humains

- presque hors contexte
- yes do séquelles

irrégulier  
Type 3

langage  
informel

Tongues  
humaines

langues humaines  
par régularisation :

Pierre

que Marie

que Valerie

connaît

regarde

connaît

Sophie

( { [ { ) ] }

( { [ ] } )

a b c c b a

pas régularisé

Mélie

Pierre et Valérie

sont les pectivants

une étudiante, un pizzaiolo et une serveuse



pas hors contexte

suivi vertes  
complétives

suisse allemand néerlandais

$S_1 S_2 S_3 V_1 V_2 V_3$

Propriétés de les grammaires  
hors contexte suffisent

Analyse syntaxique

des CFG (context-free grammar)

(automates à pile  
pushdown) certains états

→ pas déterministe

reconnaissance

NP-compl.

- algo<sup>D</sup> cubique ( $O(n^3)$ ) où  $n$  est le nombre de mots de la phrase)

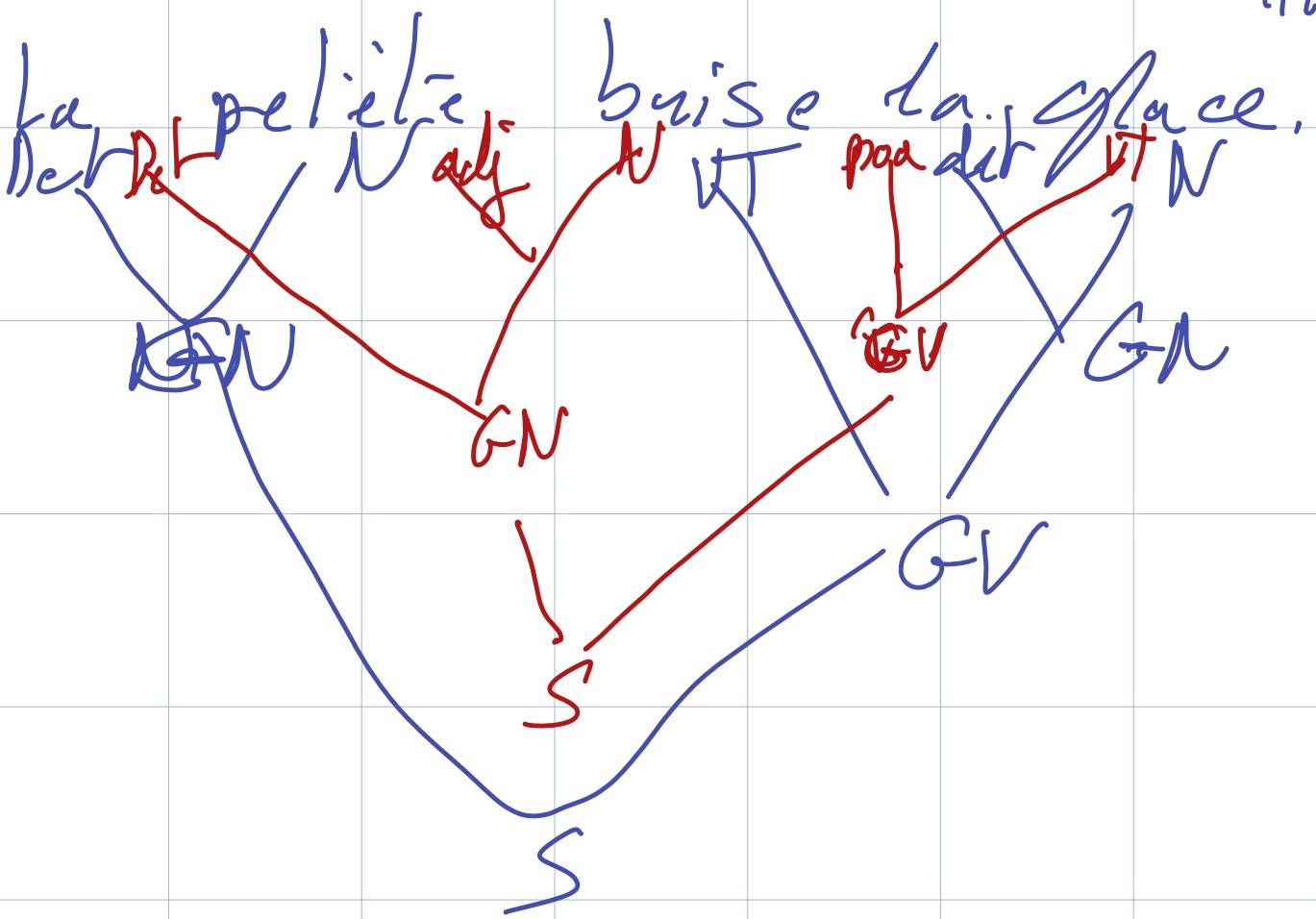
cste multiplicative  
 $O(G)$ ) taille de la  
grammaire

- programmation dynamique:

idée n'augmente un peu  
la complexité en espace

pour économiser en temps  
 $n^3$ : cubique polynomial

HMM



$S \rightarrow GN\ GV$

$GV \rightarrow VT\ GN$  |  $\text{Prodir}\ VT$

$GN \rightarrow \text{Det AdjA N} \mid \text{Det N}$

$S \rightarrow GN\text{ }GV$

$GN \rightarrow VT\text{ }GN$  | Prof VT  
 $GN \rightarrow Det\text{ }AdjA\text{ }N$  | Det N

$Det \rightarrow la$

$Prof \rightarrow la$

$VT \rightarrow brise$  | glace

$N \rightarrow petite$  | glace ) brise  
 $AdjA \rightarrow petite$

Algo étape 1  
mettre la grammaire  
en forme normale de Chomsky

On va donc faire une grammaire  $G$   
en une grammaire  $G'$   
telle que  $L(G) = L(G')$   
et qui n'a que des règles

$X \rightarrow Y Z$

un NonTerminal se décompose en 2 NonTerminaux

ou

$X \rightarrow M$

{ un NonTerminal  
donne 1 terminal

$S \rightarrow GN\ GV$

$GV \rightarrow VT\ GN$

$GN \rightarrow \text{Det } \text{AdjA } N$

$\text{Det} \rightarrow la$

$\text{ProA} \rightarrow la$

$VT \rightarrow brise$

$N \rightarrow petite$

$\text{AdjA} \rightarrow petite$

$\text{ProA} \rightarrow VT$

$\text{Dct} \rightarrow N$

$glace$

$glace$

$) brise$

Si on a une règle qui  
contient un terminal et  
d'autre terminaux ou non terminaux  
si une règle est

$$X \rightarrow a X b Y$$

on prend des nouveaux  
non terminaux  $\langle a \rangle$   $\langle b \rangle$   
on transforme la règle s

$$X \rightarrow \langle a \rangle X \langle b \rangle Y$$
$$\langle a \rangle \rightarrow a$$
$$\langle b \rangle \rightarrow b$$

et on ajoute les règles

En partant d'une CFG quelque  
on peut arriver à une CFG  
qui reconnaît le même langage  
avec des règles)

•  $X \rightarrow a \in T$

•  $X \rightarrow W X Z$

que des  
non terminaux

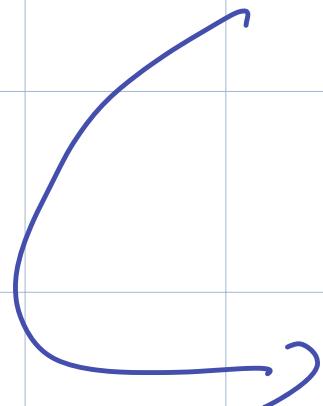
$$X \rightarrow a \neq b \neq y$$

)

$$\begin{aligned} X &\rightarrow X(a)X(b)Y \\ \langle a \rangle &\rightarrow a \\ \langle b \rangle &\rightarrow b \end{aligned}$$

on élimine les règles qui donnent 3 au plus NT ainsi

$$X \rightarrow XYZT$$



deux nouvelles NT  
 $\langle XY \rangle$  et  $\langle ZT \rangle$

$$X \rightarrow \langle XY \rangle \langle ZT \rangle$$

$$\begin{aligned} \langle XY \rangle &\rightarrow XY \\ \text{INT } \langle ZT \rangle &\rightarrow ZT \end{aligned}$$

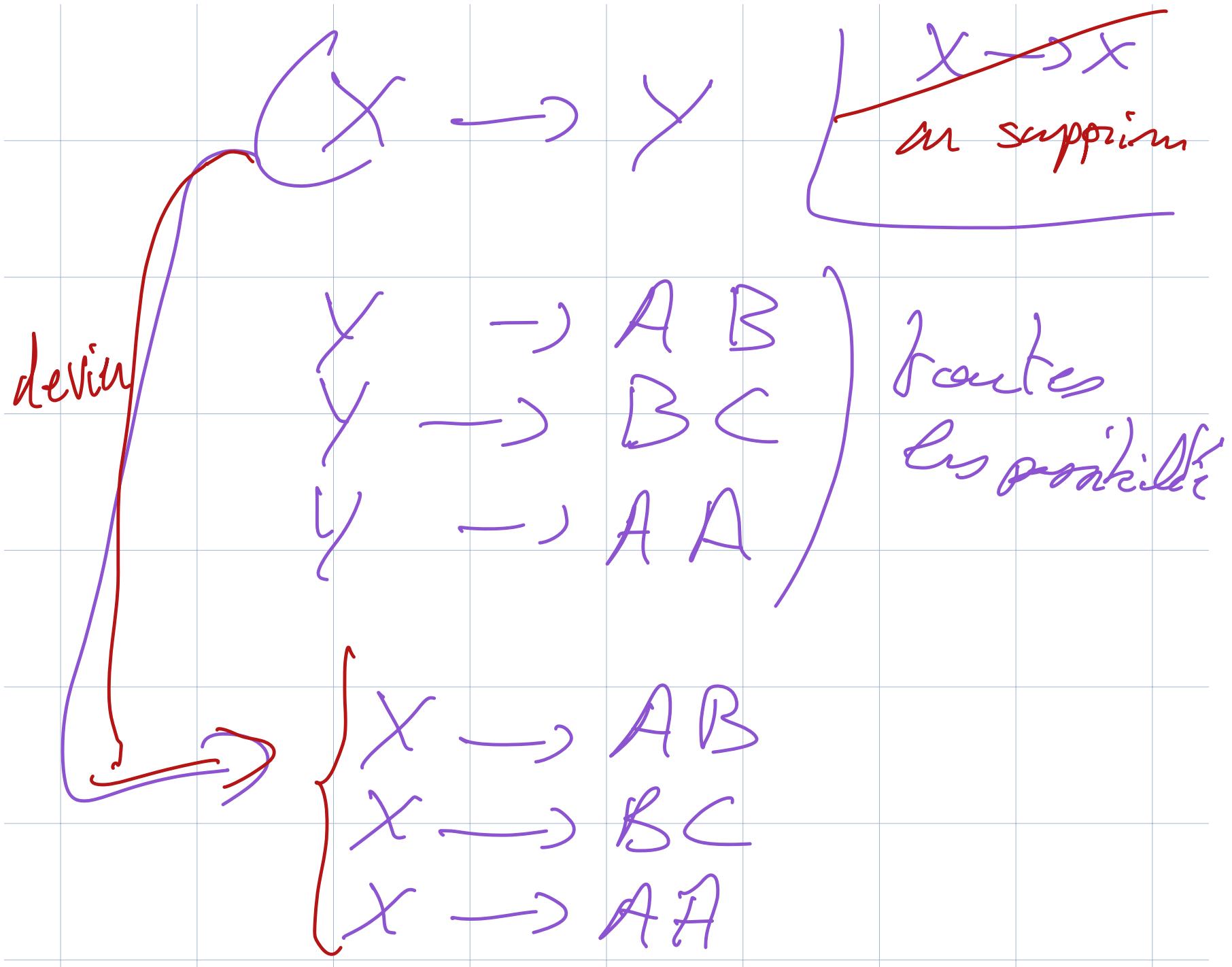
$X \rightarrow A B C D E$

$X \rightarrow \langle A B C \rangle \langle D E \rangle$

$\langle A B C \rangle \rightarrow (A B) C$

$\langle A B \rangle \rightarrow A B$

$\langle D E \rangle \rightarrow D E$



Grammaire en forme normale de Chomsky

$$X \rightarrow Y Z$$

INT                  2NT

( $X, Y, Z$   
sont des NT  
 $X = Y$  possible  
 $Y = Z = -$ )

ou

$$X \rightarrow m$$

INT                  Terminal.

$S \rightarrow GN\ GV$

$GV \rightarrow VT\ GN$

$GN \rightarrow Det \langle AdjA\ N \rangle$

$| ProA\ VT$

$| Det\ N$

$Det \rightarrow la$

$ProA \rightarrow la$

$VT \rightarrow brise$

$| glace$

$N \rightarrow petite$

$| glace$

$) brise$

$AdjA \rightarrow petite$

$\langle AdjA\ N \rangle \rightarrow AdjA\ N$

la	petit·e·	brié·	la	glace
dct	n	vt	dct	n
proa	adjf	n	proA	vt
	GN	(Adj A N)	Ø	GN GV

