NGUYEN Thi Minh Tuyen

Théorie des automates et langages formels

Grammaires Non Contextuelles

Langage et compilateur

- Langages de programmation
- Compilation d'un programme: une opération qui génère un programme équivalent en langage machine.
- 2 phases:
  1. Analyse syntaxique 🡐
  2. Traduction en langage machine

Exemple: EA (Expressions Arithmétiques)

- **Règle 1**: Tout nombre appartient à EA
- **Règle 2**: Si x et y appartiennent à EA, alors

(x) -(x) (x+y) (x-y) (x*y)

appartiennent aussi à EA.

Une autre façon pour définir les EAs : utiliser une liste de règles de substitutions similaires à des règles grammaticales.

Exemple : Règles de substitutions

Règles de substitutions qui définissent les EA:

```
S  → EA
EA → (EA + EA)
EA → (EA−EA)
EA → (EA∗EA)
EA → (EA)
EA → −(EA)
EA → NOMBRE
```

Comment définir NOMBRE?

<u>NOMBRE</u> → <u>PREMIER-CHIFFRE</u>

<u>PREMIER-CHIFFRE</u> → <u>PREMIER-CHIFFRE</u> <u>AUTRE-CHIFFRE</u>

<u>PREMIER-CHIFFRE</u> → 1 2 3 4 5 6 7 8 9

<u>AUTRE-CHIFFRE</u> → 0 1 2 3 4 5 6 7 8 9

Exemple : générer un nombre

$S \Rightarrow EA \Rightarrow (EA*EA) \Rightarrow ((EA+EA)*EA)$

$\Rightarrow ((EA+EA)*(EA+EA)) \ldots \Rightarrow ((3+4)*(6+7))$

Comment générer le nombre 1066?

NOMBRE ⇒PREMIER-CHIFFRE

⇒ PREMIER-CHIFFRE AUTRE-CHIFFRE

⇒ PREMIER-CHIFFRE AUTRE-CHIFFRE AUTRE-CHIFFRE

⇒ PREMIER-CHIFFRE AUTRE-CHIFFRE AUTRE-CHIFFRE AUTRE-CHIFFRE

⇒ 1 0 6 6

EA:

S → EA

EA →(EA + EA)

EA →(EA–EA)

EA →(EA*EA)

EA →(EA)

EA →–(EA)

EA → NOMBRE

NOMBRES

NOMBRE	→	PREMIER-CHIFFRE
PREMIER-CHIFFRE	→	PREMIER-CHIFFRE AUTRE-CHIFFRE
PREMIER-CHIFFRE	→	1 2 3 4 5 6 7 8 9
AUTRE-CHIFFRE	→	0 1 2 3 4 5 6 7 8 9

Définition

Une grammaire non contextuelle est définie par:

1. Un alphabet Σ de lettres. On appelle un élément de Σ un terminal (Σ est l'alphabet des terminaux).
2. Un ensemble de symboles, appelés "non-terminaux ou variables". Cet ensemble doit inclure un symbole particulier S, dit symbole de départ ou initial.
3. Un ensemble fini de productions de la forme:

$$A \rightarrow a$$

où A est un symbole non-terminal et

a est une suite finie de variables (non-terminaux) et de terminaux.

Exemples

Terminaux : (,), +, -, *, nombres
Non-terminaux : S, EA

EA:
S → EA
EA → (EA + EA)
EA → (EA–EA)
EA → (EA*EA)
EA → (EA)
EA → –(EA)
EA → NOMBRE

NOMBRES:
S → PREMIER-CHIFFRE
PREMIER-CHIFFRE → PREMIER-CHIFFRE AUTRE-CHIFFRE
PREMIER-CHIFFRE → 1 2 3 4 5 6 7 8 9
AUTRE-CHIFFRE → 0 1 2 3 4 5 6 7 8 9

Terminaux : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
Non-terminaux : S, PREMIER-CHIFFRE, AUTRE-CHIFFRE

Exemples

$S \rightarrow aS \qquad S \rightarrow \Lambda$

$$
\begin{aligned}
S &\Rightarrow aS \\
&\Rightarrow aaS \\
&\Rightarrow aaaS \\
&\Rightarrow aaaaS \\
&\Rightarrow aaaaaS \\
&\Rightarrow aaaaaaS \\
&\Rightarrow aaaaaa\Lambda = aaaaaa
\end{aligned}
$$

Langage engendré:
$\{\Lambda, a, aa, aaa, \ldots\}$ = langage(a^*).

Exemple

$S \rightarrow SS \qquad S \rightarrow a \qquad S \rightarrow \Lambda$

$$
\begin{aligned}
S &\Rightarrow SS \\
&\Rightarrow SSS \\
&\Rightarrow SaS \\
&\Rightarrow SaSS \\
&\Rightarrow \Lambda aSS \\
&\Rightarrow \Lambda aaS \\
&\Rightarrow \Lambda aa\Lambda = aa
\end{aligned}
$$

Langage engendré:
$\{\Lambda, a, aa, aaa, \ldots\}$ = langage(a^*).

(Un nombre infini de dérivations pour le mot aa)

<u>En général</u>: variables : en majuscule

terminaux : en minuscule

Le mot vide Λ n'est ni une variable ni un terminal

Λ est effaçable.

Une variable N est effaçable s'il y a une production de type N →Λ

Exemple

$S \rightarrow aS \quad S \rightarrow bS \quad S \rightarrow a \quad S \rightarrow b$

$S \Rightarrow aS \Rightarrow abS \Rightarrow abbS \Rightarrow abba$

$S \rightarrow X \quad S \rightarrow Y \quad X \rightarrow \Lambda$

$Y \rightarrow aY \quad Y \rightarrow bY \quad Y \rightarrow a \quad Y \rightarrow b$

$S \rightarrow aS \quad S \rightarrow bS \quad S \rightarrow a \quad S \rightarrow b \quad S \rightarrow \Lambda$

$S \Rightarrow aS \Rightarrow abS \Rightarrow abbS \Rightarrow abbaS \Rightarrow abba$

$S \rightarrow aS \quad S \rightarrow bS \quad S \rightarrow \Lambda$

Exemple

S → XaaX   X → aX   X → bX   X → Λ

S ⇒XaaX ⇒ aXaaX ⇒ abXaaX ⇒ abXaabX ⇒ abaab

Combien de dérivations possibles pour le mot **baabaab**?

S → XY

X → aX

X → bX

X → a

Y → Ya

Y → Yb

Y → a

Abréviation:

S → XY

X → aX | bX | a

Y → Ya | Yb | a

Exemple

$S \rightarrow SS \mid ES \mid SE \mid \Lambda \mid DSD$

$E \rightarrow aa \mid bb$

$D \rightarrow ab \mid ba$

PAIR-PAIR = langage([aa + bb + (ab + ba)(aa+bb)*(ab + ba)]*)

$S \rightarrow aSb \mid \Lambda$

$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaaSbbb \Rightarrow aaaaSbbbb$

$\Rightarrow aaaaaSbbbbb \Rightarrow aaaaabbbbb$

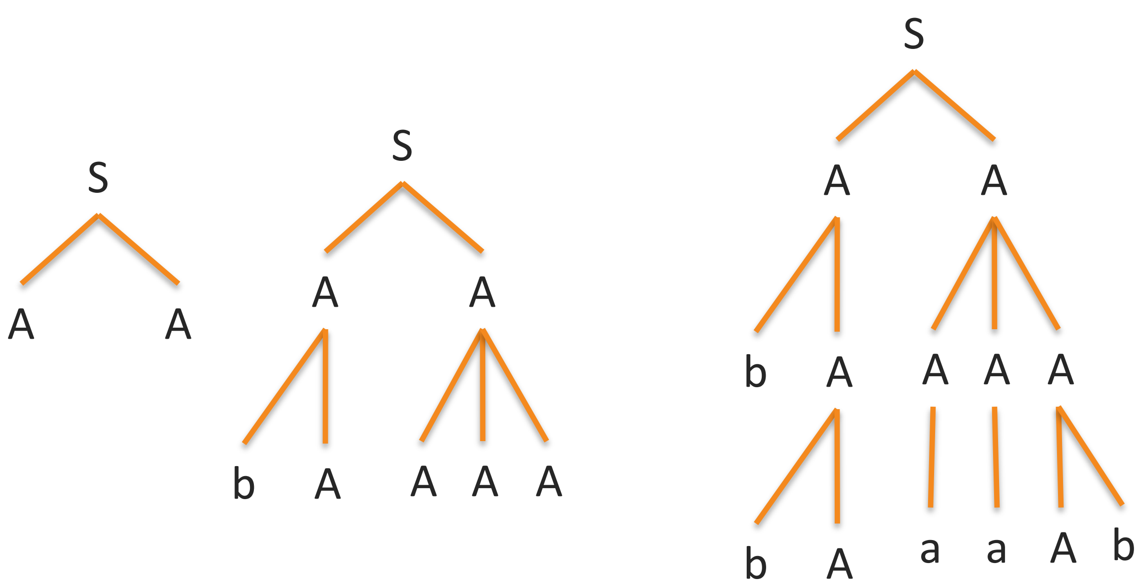
$S \rightarrow aSa \mid bSb \mid \Lambda$

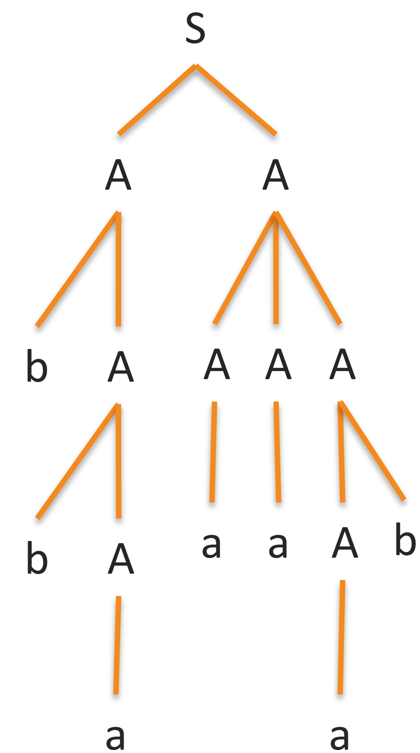
Arbres de dérivation

S → AA

A → AAA | bA | Ab | a

Arbres de dérivation du mot bbaaaab





bbaaaab

Arbre de dérivation

On dira arbre de dérivation, ou arbre de syntaxe, ou arbre de production, ou arbre de génération.

<u>Remarque</u>: Dans un arbre de dérivation chaque nœud interne est étiqueté par une variable (non-terminal) et chaque feuille est étiquetée par un terminal.

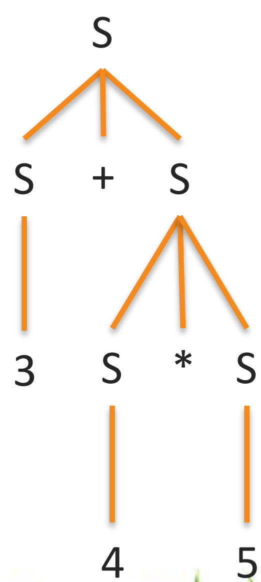
Exemple

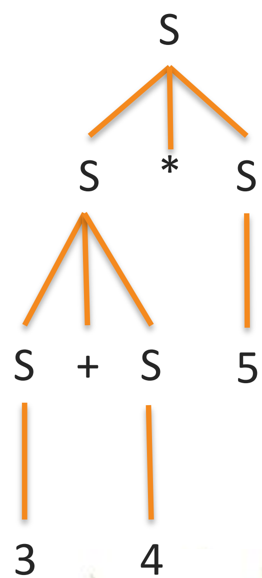
$S \rightarrow (S+S) \mid (S*S) \mid NOMBRE$

$NOMBRE \rightarrow \ldots$

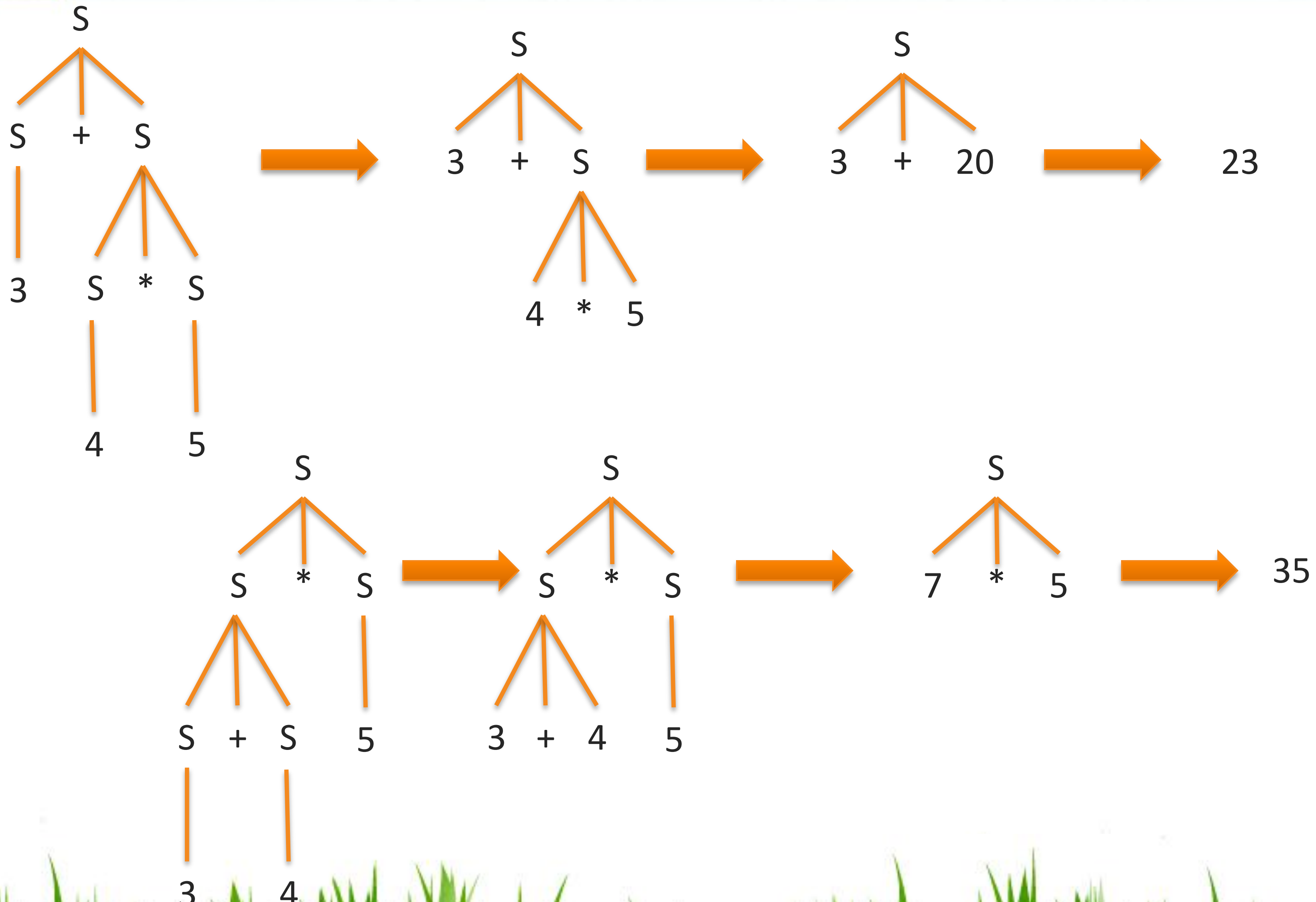
$$S \Rightarrow (S+S) \Rightarrow (S+(S*S)) \Rightarrow \ldots \Rightarrow (3+(4*5))$$

$$S \Rightarrow (S*S) \Rightarrow (S+S)*S) \Rightarrow \ldots \Rightarrow ((3+4)*5)$$

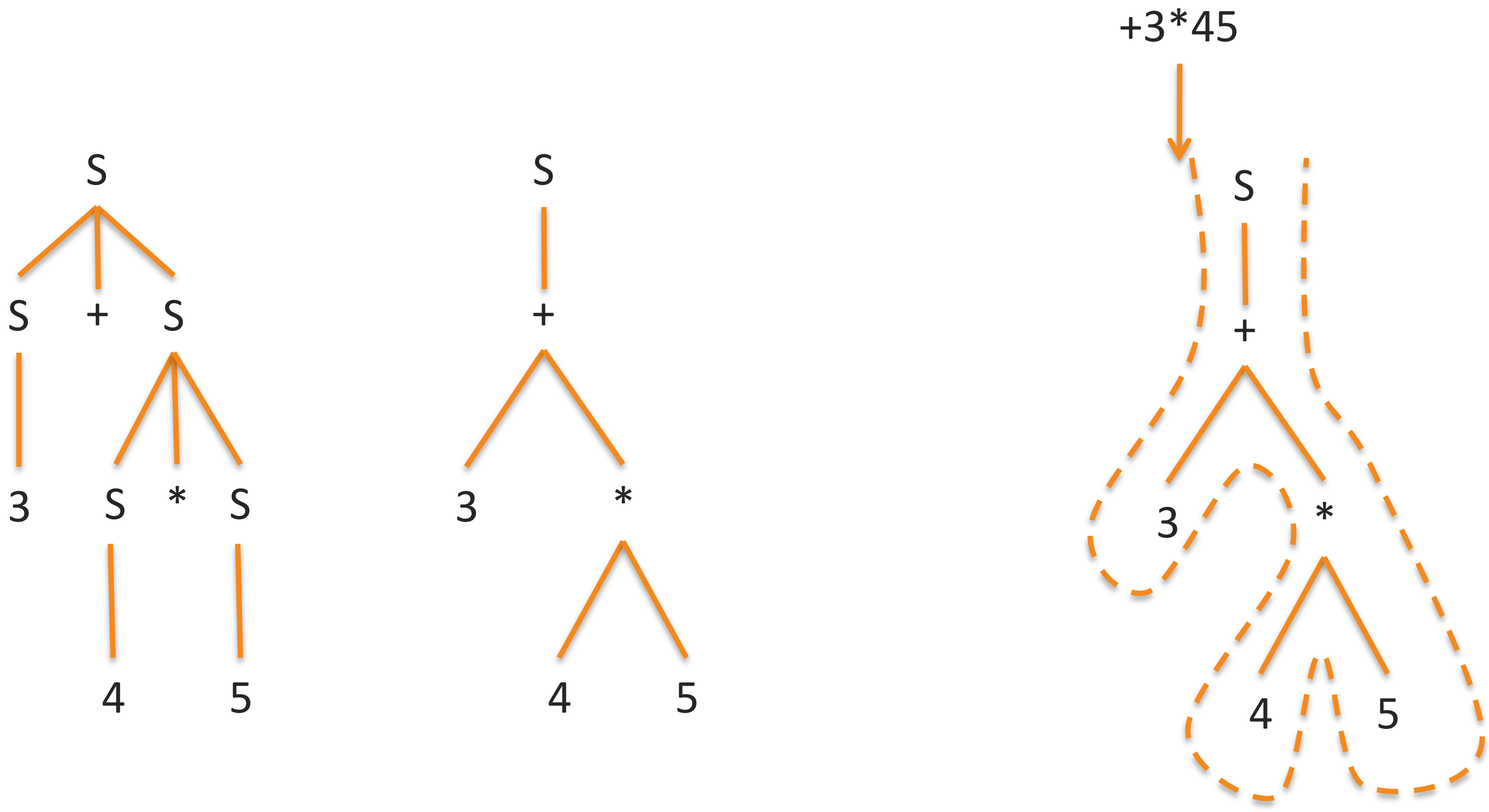




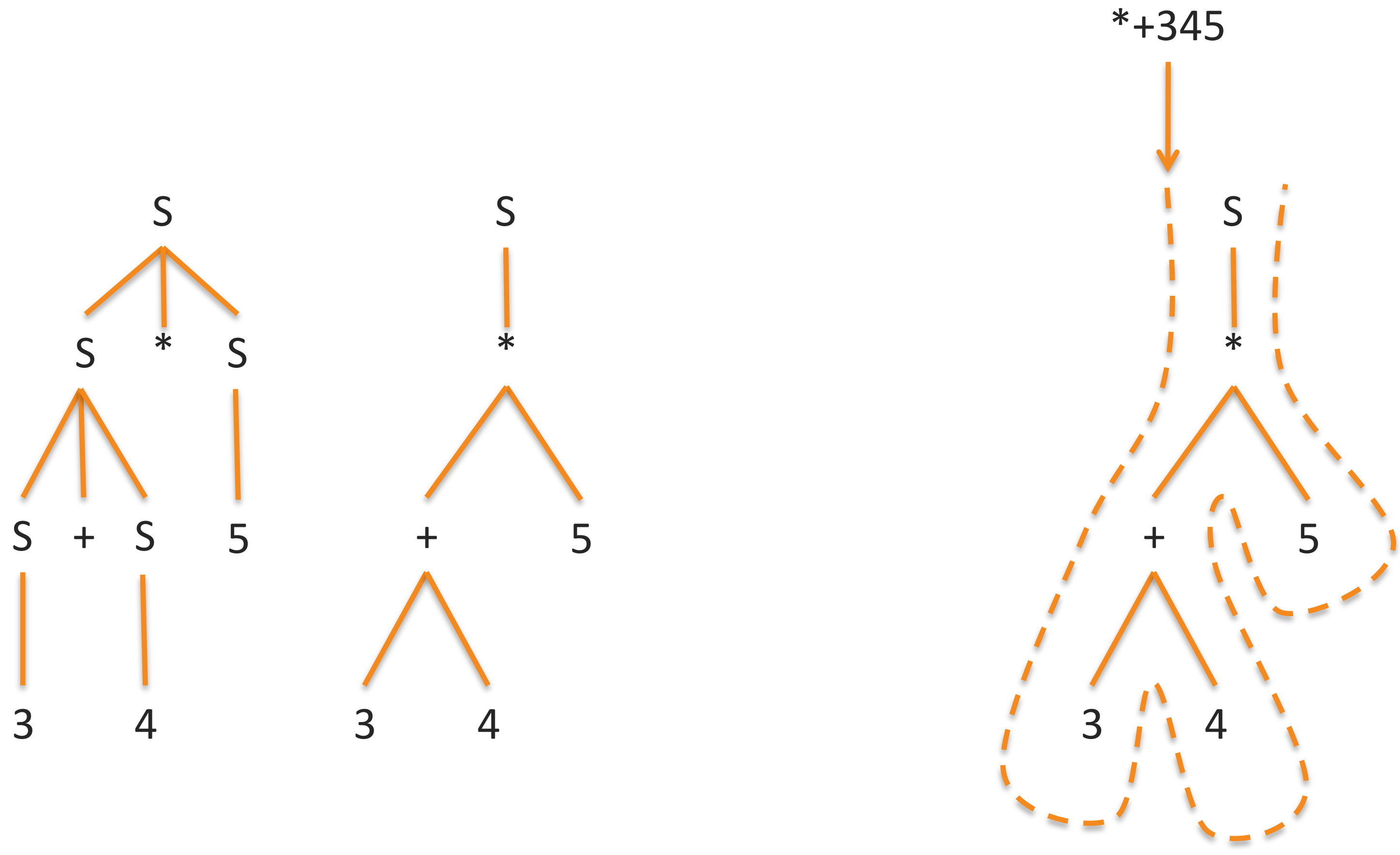
Exemple



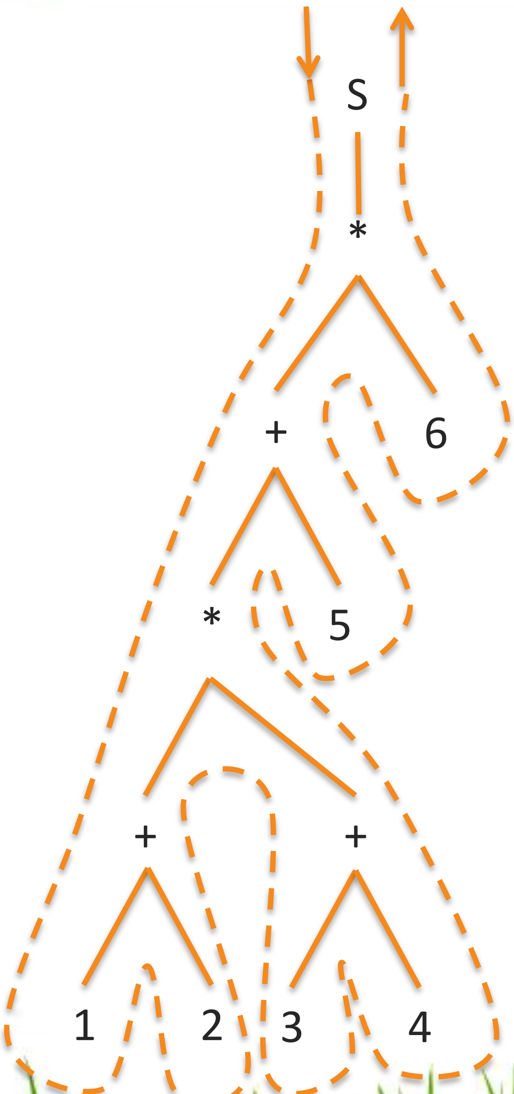
Notation de Lukasiewicz [1]



Notation de Lukasiewicz [2]



Notation de Lukasiewicz [3]



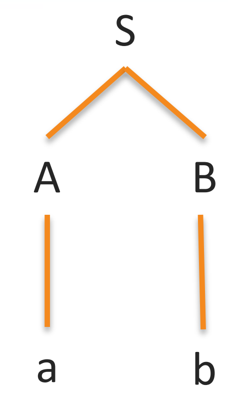
* + * + 1 2 + 3 4 5 6	+ 1 2
* + * 3 + 3 4 5 6	+ 3 4
* + * 3 7 5 6	* 3 7
* + 21 5 6	+ 21 5
* 26 6	* 26 6
156	

Ambiguité

Exemple: $S \rightarrow AB$ $\quad A \rightarrow a \quad B \rightarrow b$

$$S \Rightarrow AB \Rightarrow aB \Rightarrow ab$$

$$S \Rightarrow AB \Rightarrow Ab \Rightarrow ab$$

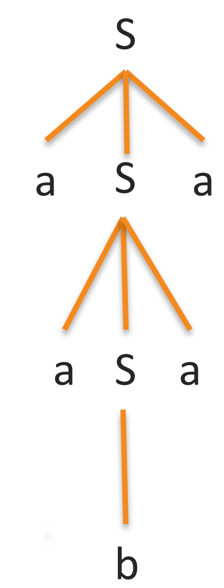


- *Une grammaire non contextuelle est* ***ambigu*** *s'il existe un mot* ***w*** *dans le langage, tel que* ***w*** *a <u>aux moins deux</u> arbres de dérivation différents.*

Exemple: $S \rightarrow aSa \quad S \rightarrow bSb$

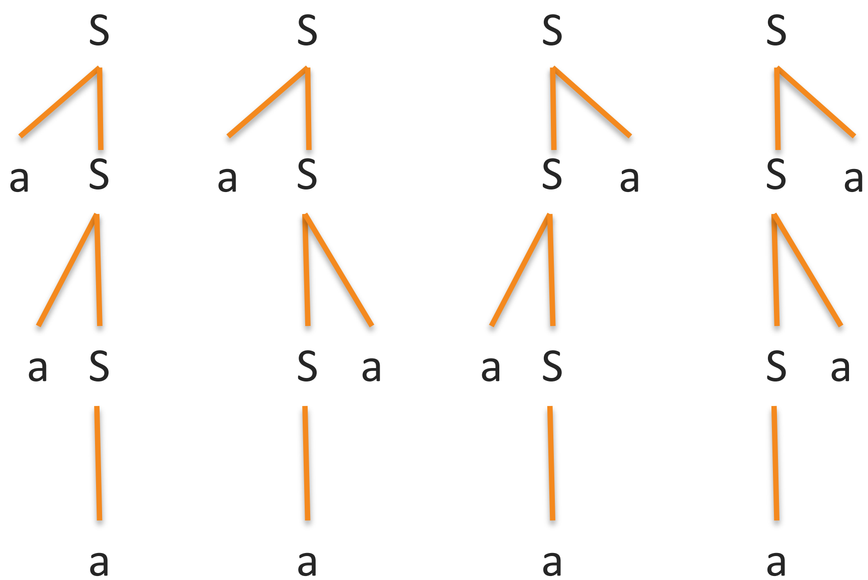
$S \rightarrow a \quad S \rightarrow b \quad S \rightarrow \Lambda$

$$S \Rightarrow aSa \Rightarrow aaSaa \Rightarrow aabaa$$



Example: Langage(a+)

$S \rightarrow aS \mid Sa \mid a$

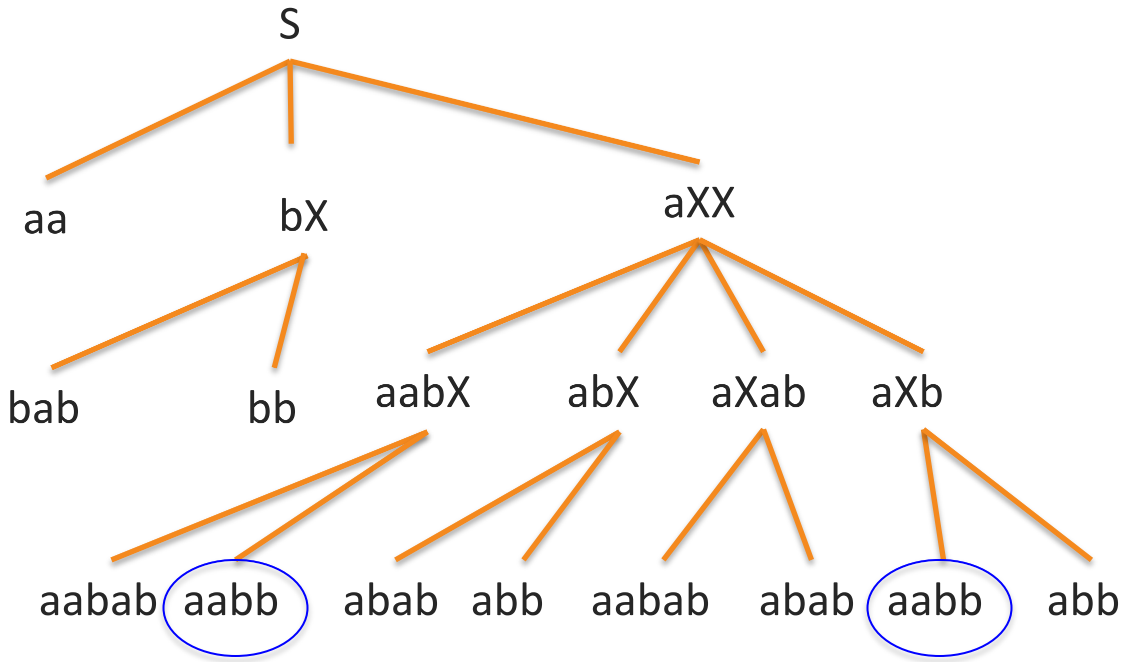


$S \rightarrow aS \mid a$

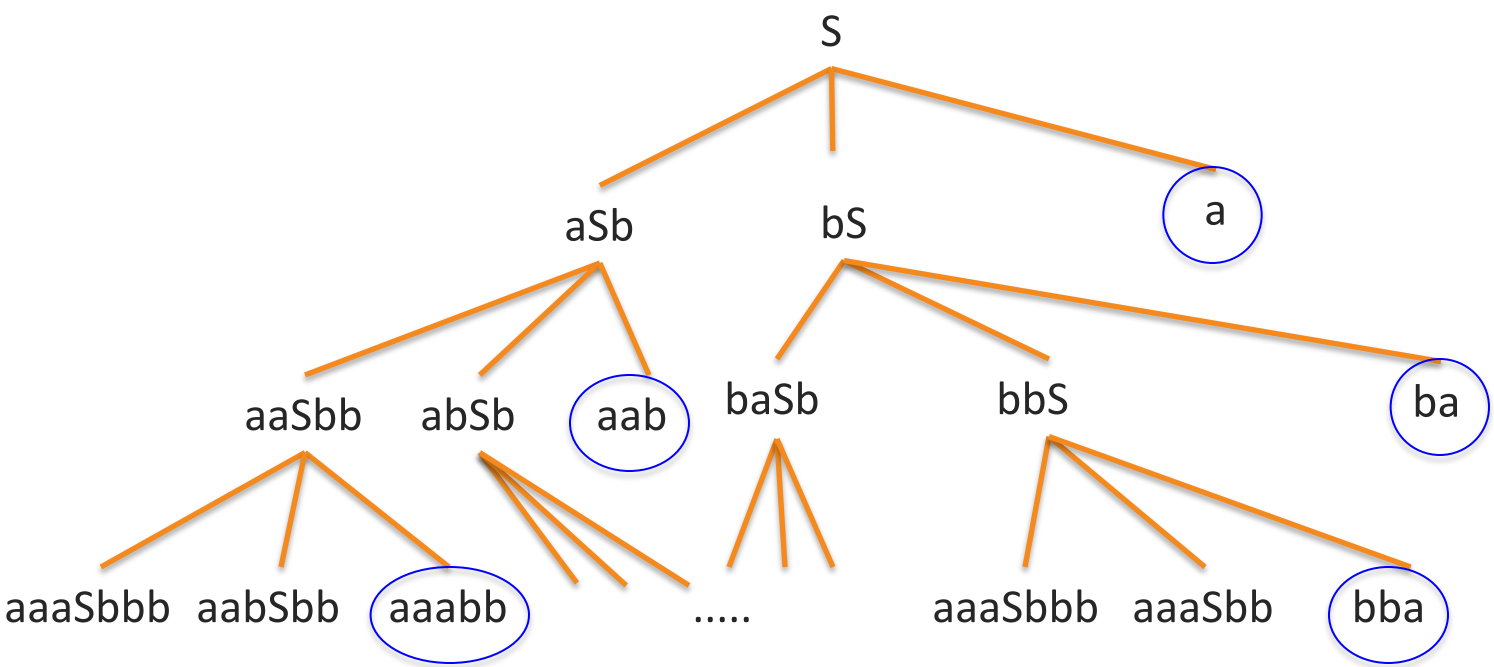
Arbre total du langage

S →aa | bX | aXX

X → ab | b

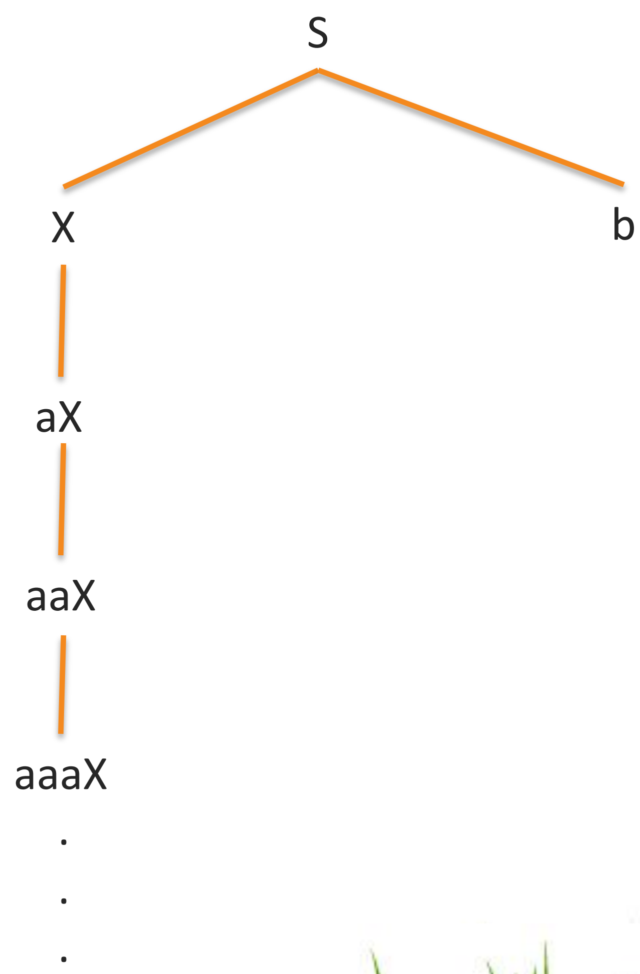


$S \rightarrow aSb \mid bS \mid a$



$S \rightarrow X \mid b$

$X \rightarrow aX$



Question?