Elimination of Left Recursion (LR)

Course Name: Compiler Design
Course Code: CSE331
Level:3, Term:3
Department of Computer Science and Engineering
Daffodil International University

Left and Right Recursive Grammars

In a context-free grammar G, if there is a production in the form $X \rightarrow Xa$ where X is a non-terminal and 'a' is a string of terminals, it is called a left recursive production. The grammar having a left recursive production is called a left recursive grammar.

And if in a context-free grammar G, if there is a production is in the form $X \rightarrow aX$ where X is a non-terminal and 'a' is a string of terminals, it is called a right recursive production. The grammar having a right recursive production is called a right recursive grammar.

Left Recursion (LR)

$A \longrightarrow A\ \alpha \mid \beta$

Right Recursion (RR)

$A \longrightarrow \alpha\ A \mid \beta$

Left Recursion (LR)

$A \longrightarrow A\,\alpha \mid \beta$

Right Recursion (RR)

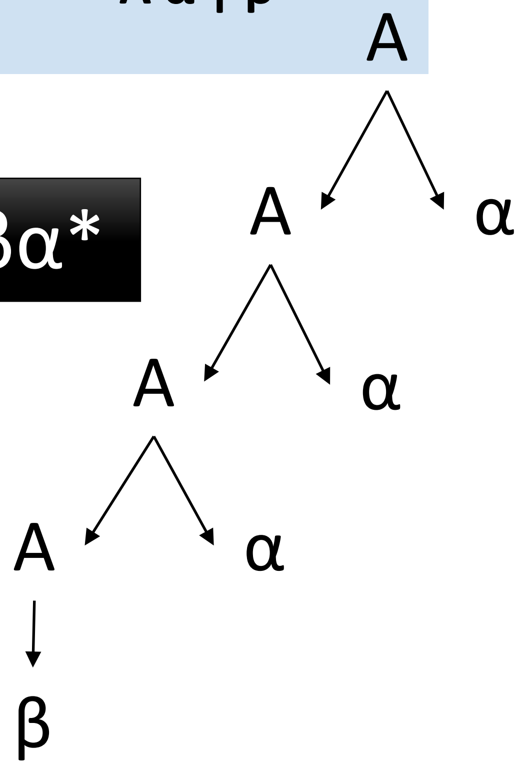
$A \longrightarrow \alpha\, A \mid \beta$

Left Recursion (LR)

$A \longrightarrow A\ \alpha\ |\ \beta$

Right Recursion (RR)

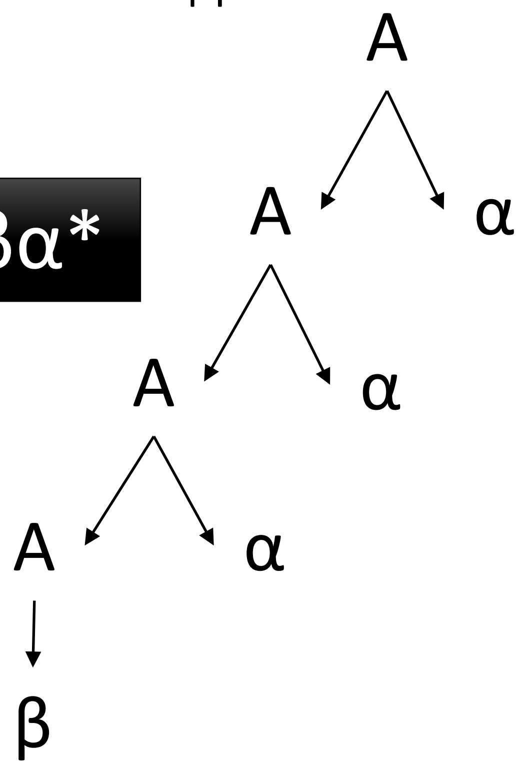
$A \longrightarrow \alpha\ A\ |\ \beta$



$\beta\alpha^*$

Left Recursion (LR)

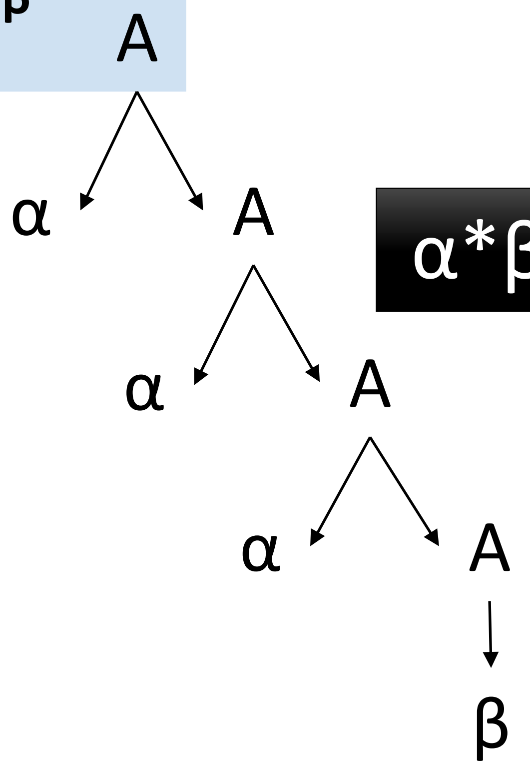
$A \longrightarrow A\ \alpha \mid \beta$



βα*

Right Recursion (RR)

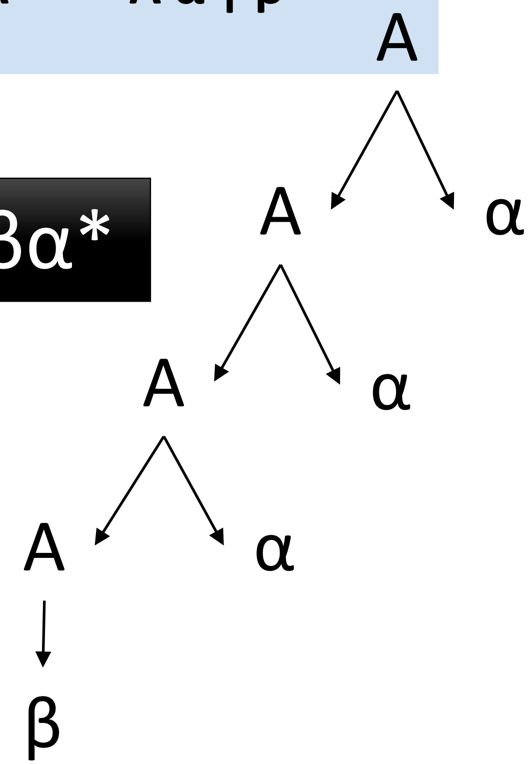
$\mathbf{A \longrightarrow \alpha\ A \mid \beta}$



α*β

Left Recursion (LR)

$A \longrightarrow A\ \alpha \mid \beta$



$\beta\alpha^*$

$A \longrightarrow A\ \alpha \mid \beta$

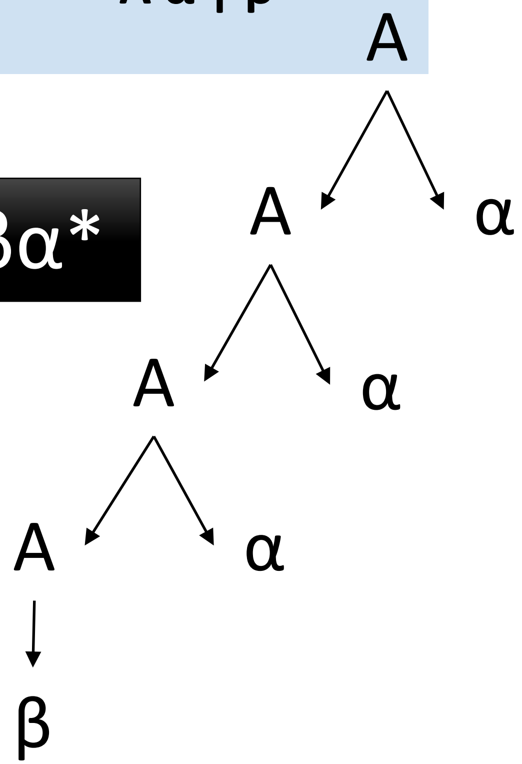
$A \longrightarrow \beta\ A'$

$A' \longrightarrow \epsilon \mid \alpha\ A'$

Left Recursion (LR)

$A \rightarrow A\ \alpha \mid \beta$

$\beta\alpha^*$



$A \rightarrow A\ \alpha \mid \beta$

$A \rightarrow \beta\ A'$

$A' \rightarrow \epsilon \mid \alpha\ A'$

Left Recursive

Example:

$E \rightarrow E + T \mid T$

Solve:

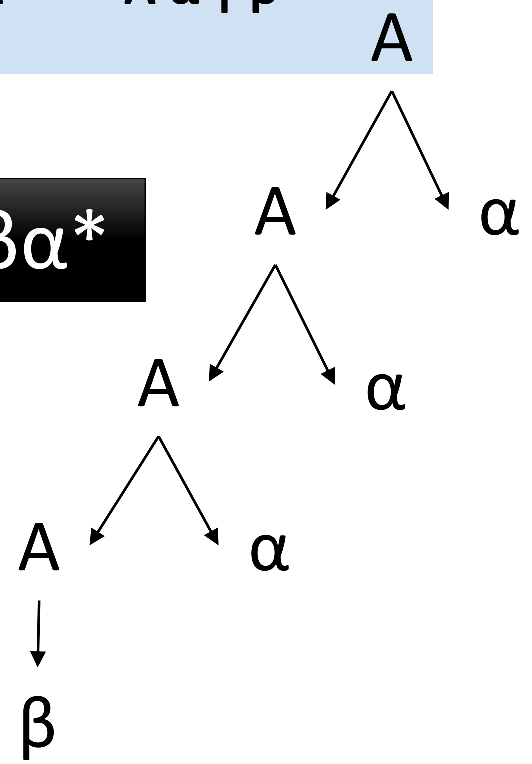
$E \rightarrow T\ E'$

$E' \rightarrow \epsilon \mid +\ T\ E'$

Left Recursion (LR)

$A \rightarrow A\ \alpha \mid \beta$

$\beta\alpha^*$



$A \rightarrow \beta\ A'$

$A' \rightarrow \epsilon \mid \alpha\ A'$

$A \rightarrow A\ \alpha \mid \beta$

Example: $E \rightarrow \boxed{E}\ \boxed{+T} \mid T$

Solve:

$E \rightarrow T\ E'$

$E' \rightarrow \epsilon \mid +\ T\ E'$

Exercise

Exercise 1: $T \rightarrow T * F \mid F$

Exercise 2: $F \rightarrow (E) \mid id$

Exercise 3: $E \rightarrow E + T \mid T$

$T \rightarrow T * F \mid F$

$F \rightarrow (E) \mid id$

Elimination of Left Factoring (LF)

Grammar

Deterministic Grammar

Non-Deterministic Grammar

$$A \longrightarrow A\,\alpha \mid \beta$$

$$A \longrightarrow \alpha\,A \mid \alpha\,\beta$$

Grammar

Deterministic Grammar

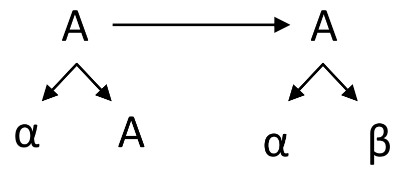
$$A \longrightarrow A\,\alpha \mid \beta$$

Non-Deterministic Grammar

$$A \longrightarrow \alpha\,A \mid \alpha\,\beta$$

$$A \longrightarrow \alpha\,A$$
$$A \longrightarrow \alpha\,\beta$$

If I want to have **"α β"**



As we can derive both **αA** & **αβ** from the grammar, both have same prefix (**α**), so, the grammar is non-deterministic.

Grammar

Deterministic Grammar

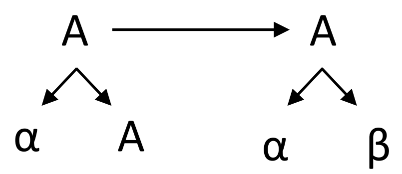
$$A \longrightarrow A\,\alpha \mid \beta$$

Non-Deterministic Grammar

$$A \longrightarrow \alpha\,A \mid \alpha\,\beta$$

$$A \longrightarrow \alpha\,A$$
$$A \longrightarrow \alpha\,\beta$$

If I want to have **"α β"**



As we can derive both **αA** & **αβ** from the grammar, both have same prefix (**α**), so, the grammar is non-deterministic.

To make this grammar deterministic, we can apply left factoring.

Grammar

Deterministic Grammar

$A \longrightarrow A\,\alpha \mid \beta$

Non-Deterministic Grammar

$A \longrightarrow \alpha\,A \mid \alpha\,\beta$

$A \longrightarrow \alpha \quad A$

$A \longrightarrow \alpha \quad \beta$

A

If I want to have **“α β”**

Grammar

Deterministic Grammar

$$A \longrightarrow A\,\alpha \mid \beta$$

Non-Deterministic Grammar

$$A \longrightarrow \alpha\, A \mid \alpha\, \beta$$

$$A \longrightarrow \alpha\ A$$

$$A \longrightarrow \alpha\ \beta$$

$$A \longrightarrow \alpha$$

If I want to have **"α β"**

Grammar

Deterministic Grammar

$$A \longrightarrow A\,\alpha \mid \beta$$

Non-Deterministic Grammar

$$A \longrightarrow \alpha\,A \mid \alpha\,\beta$$

$$A \longrightarrow \alpha\ A$$
$$A \longrightarrow \alpha\ \beta$$

$$A \longrightarrow \alpha\ A'$$

If I want to have **“α β”**

Grammar

Deterministic Grammar

$$A \longrightarrow A\,\alpha \mid \beta$$

Non-Deterministic Grammar

$$A \longrightarrow \alpha\,A \mid \alpha\,\beta$$

$$A \longrightarrow \alpha \quad A$$
$$A \longrightarrow \alpha \quad \beta$$

If I want to have **"α β"**

$$A \longrightarrow \alpha \quad A'$$

$$A' \longrightarrow A \mid \beta$$

Grammar

Deterministic Grammar

$$A \longrightarrow A\,\alpha \mid \beta$$

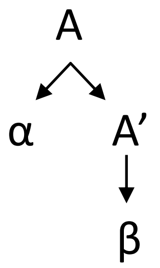
Non-Deterministic Grammar

$$A \longrightarrow \alpha\,A \mid \alpha\,\beta$$

$$A \longrightarrow \alpha\ \ A$$
$$A \longrightarrow \alpha\ \ \beta$$

If I want to have **“α β”**

$$A \longrightarrow \alpha\ \ A'$$
$$A' \longrightarrow A \mid \beta$$



$$A \longrightarrow \alpha A \mid \alpha \beta$$

$$A \longrightarrow \alpha \; A'$$

$$A' \longrightarrow A \mid \beta$$

Exercise:

$$
\begin{aligned}
S &\longrightarrow i\,E\,t\,S \\
&\quad \mid i\,E\,t\,S\,e\,S \\
&\quad \mid a \\
E &\longrightarrow b
\end{aligned}
$$

Solution:

$$
\begin{aligned}
S &\longrightarrow i\,E\,t\,S \\
&\quad \mid i\,E\,t\,S\,e\,S \\
&\quad \mid a \\
E &\longrightarrow b
\end{aligned}
$$

$$A \longrightarrow \alpha A \mid \alpha \beta$$

$$A \longrightarrow \alpha \ A'$$

$$A' \longrightarrow A \mid \beta$$

Exercise:

$$S \longrightarrow i\,E\,t\,S \mid i\,E\,t\,S\,e\,S \mid a$$

$$E \longrightarrow b$$

Left Factoring

Solution:

$$S \longrightarrow \boxed{i\,E\,t\,S} \mid \boxed{i\,E\,t\,S}\,e\,S \mid a$$

$$E \longrightarrow b$$

$$S \longrightarrow i\,E\,t\,S\ \ S' \mid a$$

$$S' \longrightarrow \epsilon \mid e\,S$$

$$E \longrightarrow b$$

Grammar is converted to deterministic.

Exercises for practice:

1

$A \longrightarrow aAB \mid aA$

$B \longrightarrow bB \mid b$

2

$E \longrightarrow T + E \mid T\,\epsilon$

3

$S \longrightarrow bSSaaS \mid bSSaSb \mid bSb \mid a$

THANK YOU