Bhita Hardasmalani
C14-58.

THADOMAL SHAHANI
TSEC
ENGINEERING COLLEGE

EXPERIMENT NO. 10.

Aim: write a program to eliminate left recursion from the given grammar.

Theory:

A production of grammar is said to have left recursion if the leftmost vaiiable of its RHS is same as vaiiable of its LHS.
A grammar G (V, T, P, S) is left recursive if it has a production in the form,

$$A \to A\alpha \mid \beta$$

the above grammar is left recursive because the left of production is only at first position on RHS.
we can replace left recursion by replacing a pair of production with

$$A \to \beta A'$$
$$A' \to \alpha A' \mid \epsilon$$

In left recursive grammar, expansion of A will generate $A\alpha$, $A\alpha\tau$, $A\alpha\alpha\alpha$, on each side causing it to enter into an infinite loop.

Eg: $E \to E+T \mid T$
$T \to T * f \mid F$
$F \to (E) \mid id$

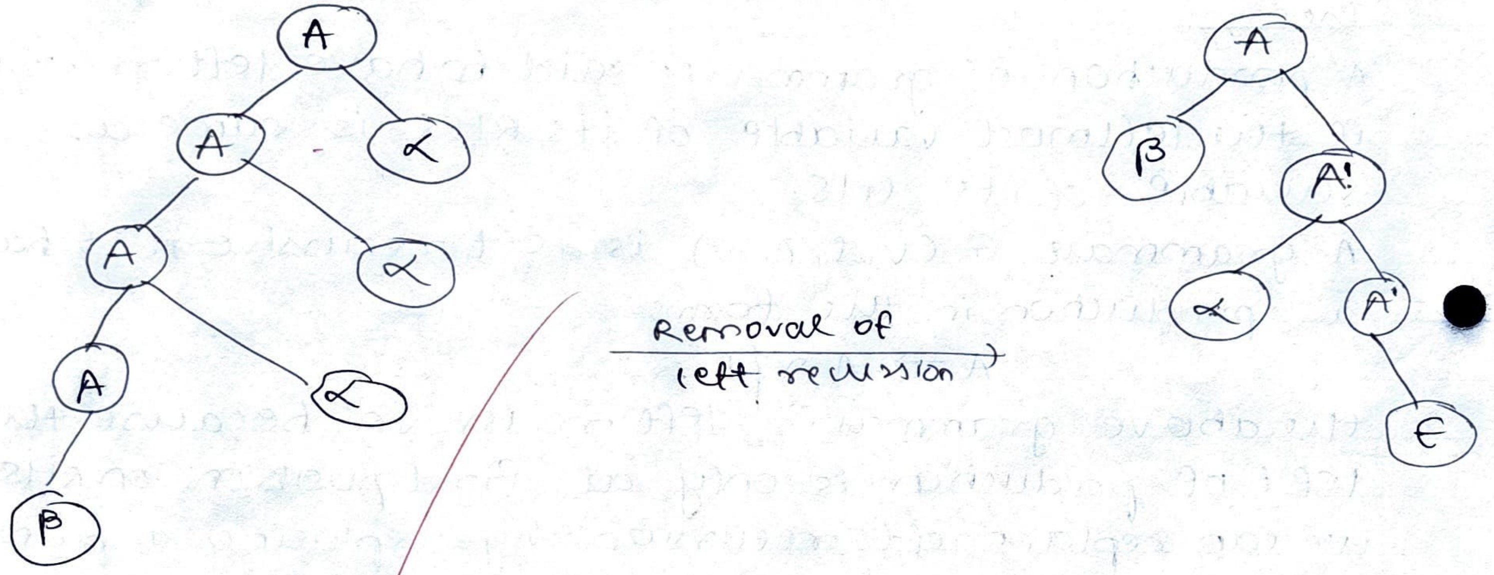
compare, $E \to E+T \mid T$
with $A \to A\alpha \mid \beta$
$\therefore A = E, \quad \alpha = +T, \quad \beta = T.$

$A \to A\alpha \mid B$ changes to $A' \to \alpha A' \mid \epsilon$
$A \to BA'$ $\quad \therefore \quad E = TE'$

$\therefore$ ~~F~~

↳ $E \to E+T \mid T$
$E \to TE'$
$E' \to +TE' \mid \epsilon$

$$A \rightarrow A\alpha \mid \beta \quad \xrightarrow[\text{left recursion}]{\text{removal of}} \quad \begin{array}{l} A \rightarrow \beta A' \\ A' \rightarrow \alpha A' \mid \epsilon \end{array}$$



compare $T \rightarrow T * F \mid F$ with $A \rightarrow A\alpha \mid \beta$

↳ $T \rightarrow FT'$

$T' \rightarrow *FT' \mid \epsilon$

∴ final set of production is.

$E \rightarrow TE'$

$E' \rightarrow +TE' \mid \epsilon$

$T \rightarrow FT'$

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

$F \rightarrow (E) \mid id$.

18/3/24

A+