CS & IT ENGINEERING

Compiler Design

Syntax Directed Translations

Lecture No. 2

By- DEVA Sir

Evaluations of SDTs

L-attributed Grammar

Computation can depend on

Parent / Left Sibling / children

Parent / Left Sibling: evaluated in top-down approach

children: evaluated in bottom-up approach

S-attributed Grammar

Computation depends only on children

(Bottom-up approach)

① $S \rightarrow S_1 S_2 \quad \{ S.x = S_1.x + S_2.x \}$

$S \rightarrow (S_1) \quad \{ S.x = S_1.x + 1 \}$

$S \rightarrow \epsilon \quad \{ S.x = 0 \}$

Q1) What is x ? ⇒ x is synthesized attribute

Q2) What is SDT definition ? ⇒ Both S-attributed & L-attributed

Q3) Find attribute value at root for $((\,)(\,))(\,)$

Q4) What is functionality of SDT ? ⇒ No. of balanced parantheses

① $S \rightarrow S_1 S_2 \quad \{ S.x = S_1.x + S_2.x \}$

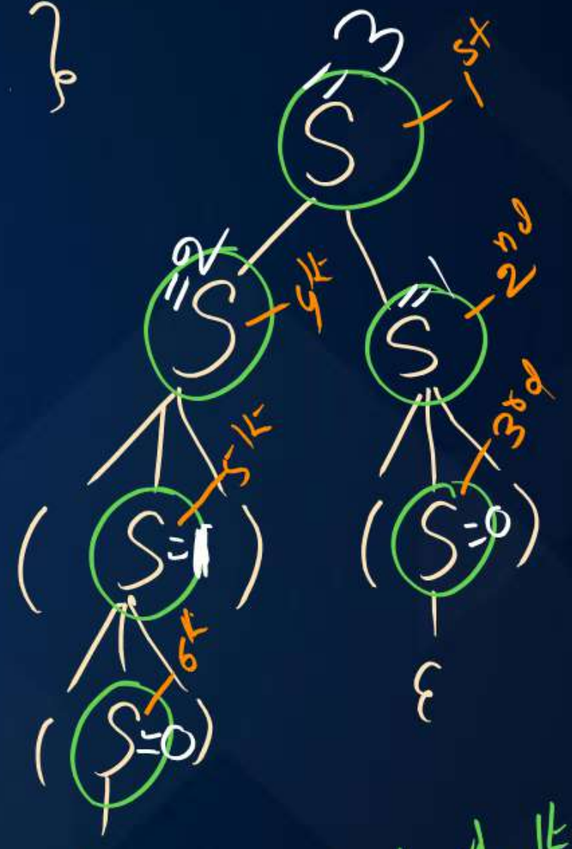
$S \rightarrow (S_1) \quad \{ S.x = S_1.x + 1 \}$

$S \rightarrow \varepsilon \quad \{ S.x = 0 \}$

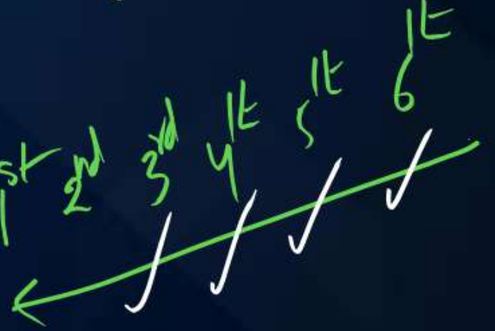
Q3 If we use S-attributed Evaluation:
(Bottom up parsing)
(Reverse of RMD)

Input: (( )) ( )

$= 3$ //



RMD Numbering:

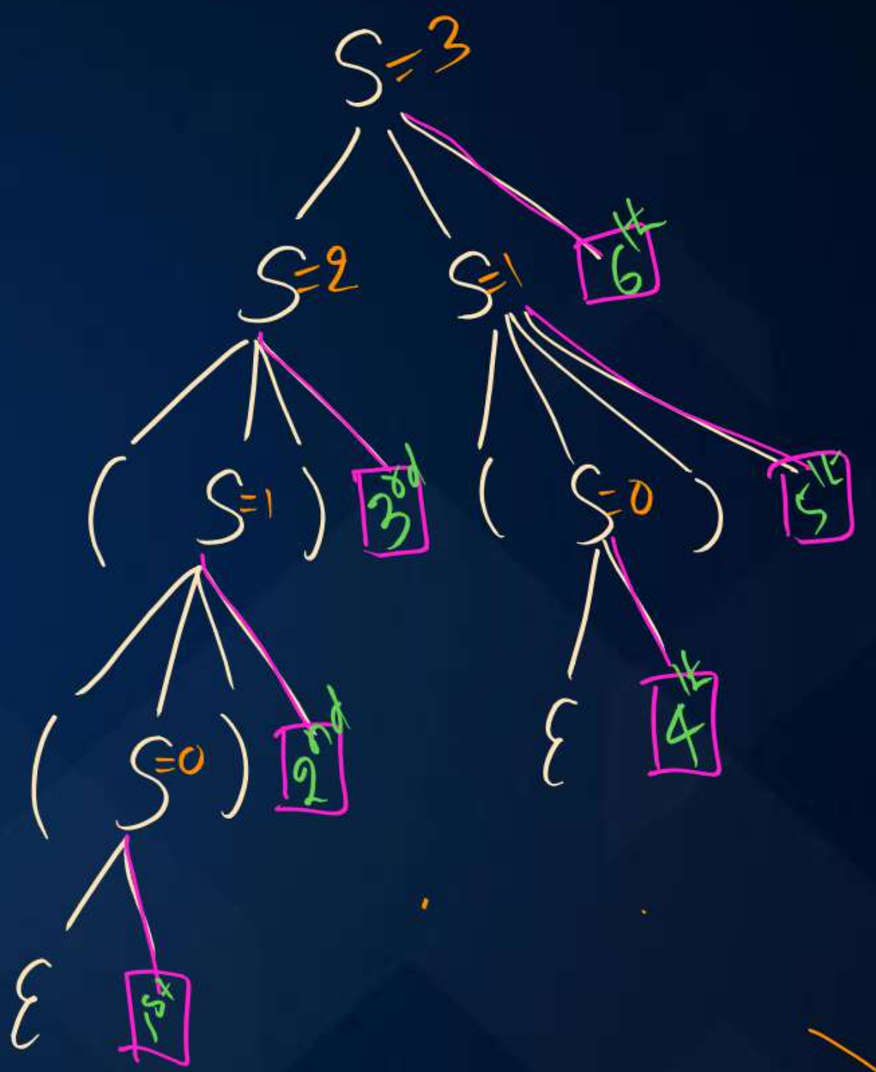


① $S \to S_1 S_2 \{S.x = S_1.x + S_2.x\}$

$S \to (S_1) \{S.x = S_1.x + 1\}$

$S \to \varepsilon \quad \{S.x = 0\}$

$= 3$



Q3 If we use L-attributed Evaluation:

(Depth Left, Left to Right)

(Inorder Approach)

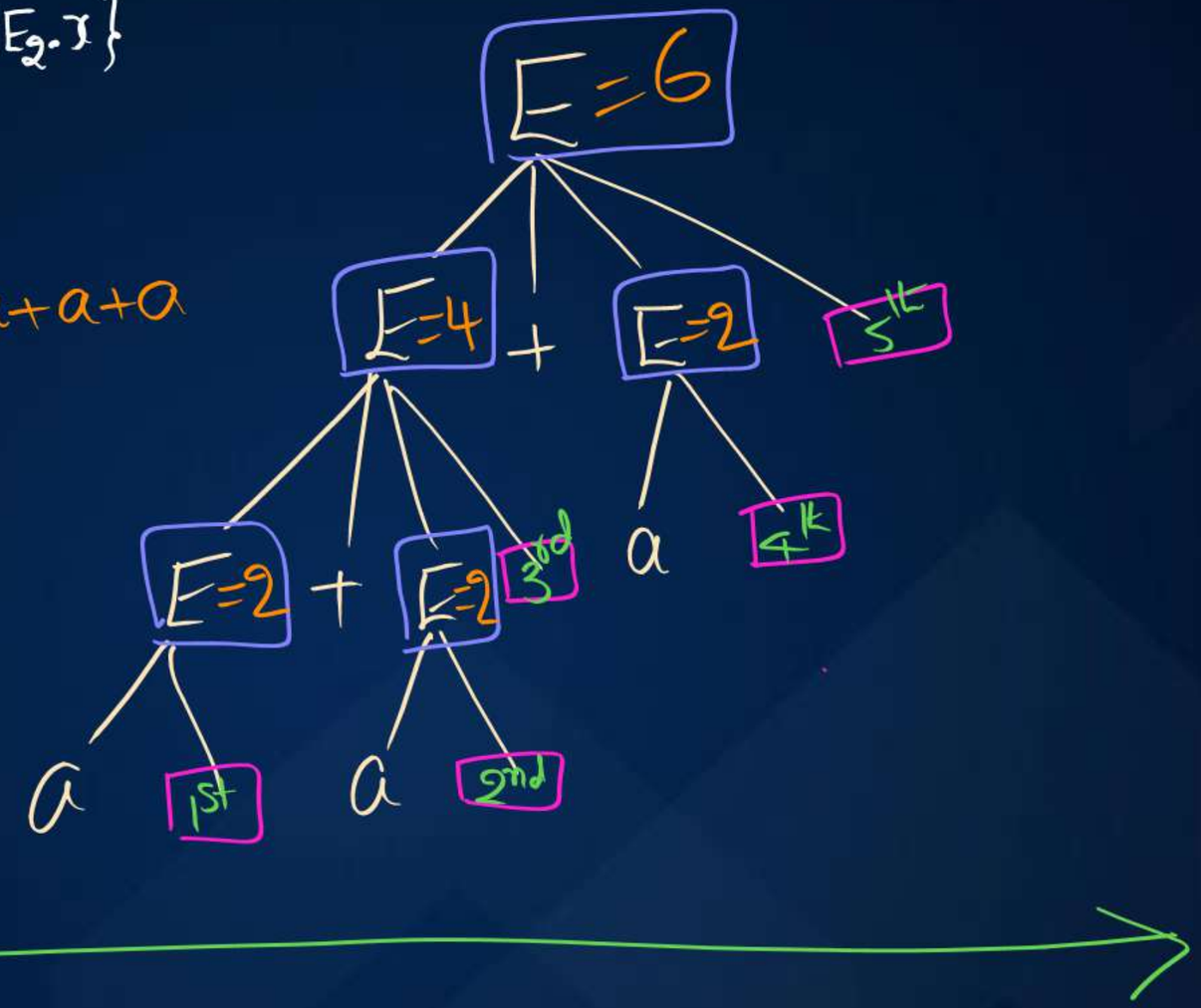
(Topological order)

→ Find translations order from Left to Right

(2) $E \rightarrow E_1 + E_2 \{E.x = E_1.x + E_2.x\}$

$E \rightarrow a \{E.x = 2\}$

Find translation for a+a+a



= 6

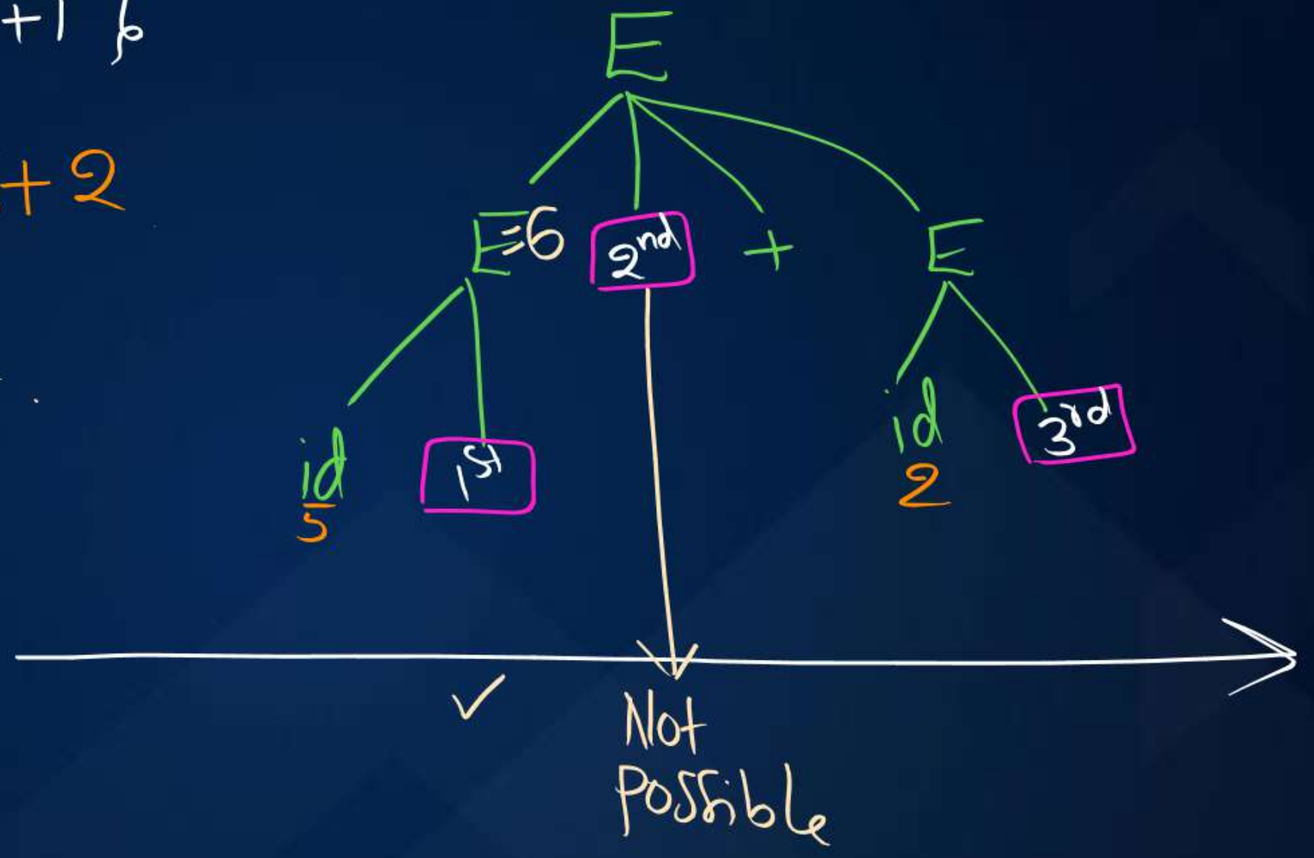
③ $E \rightarrow E_1 \{ E.x = E_1.x - E_2.x \} + E_2$

$E \rightarrow id \{ E.x = id.val + 1 \}$

Find translation for 5+2

⇓

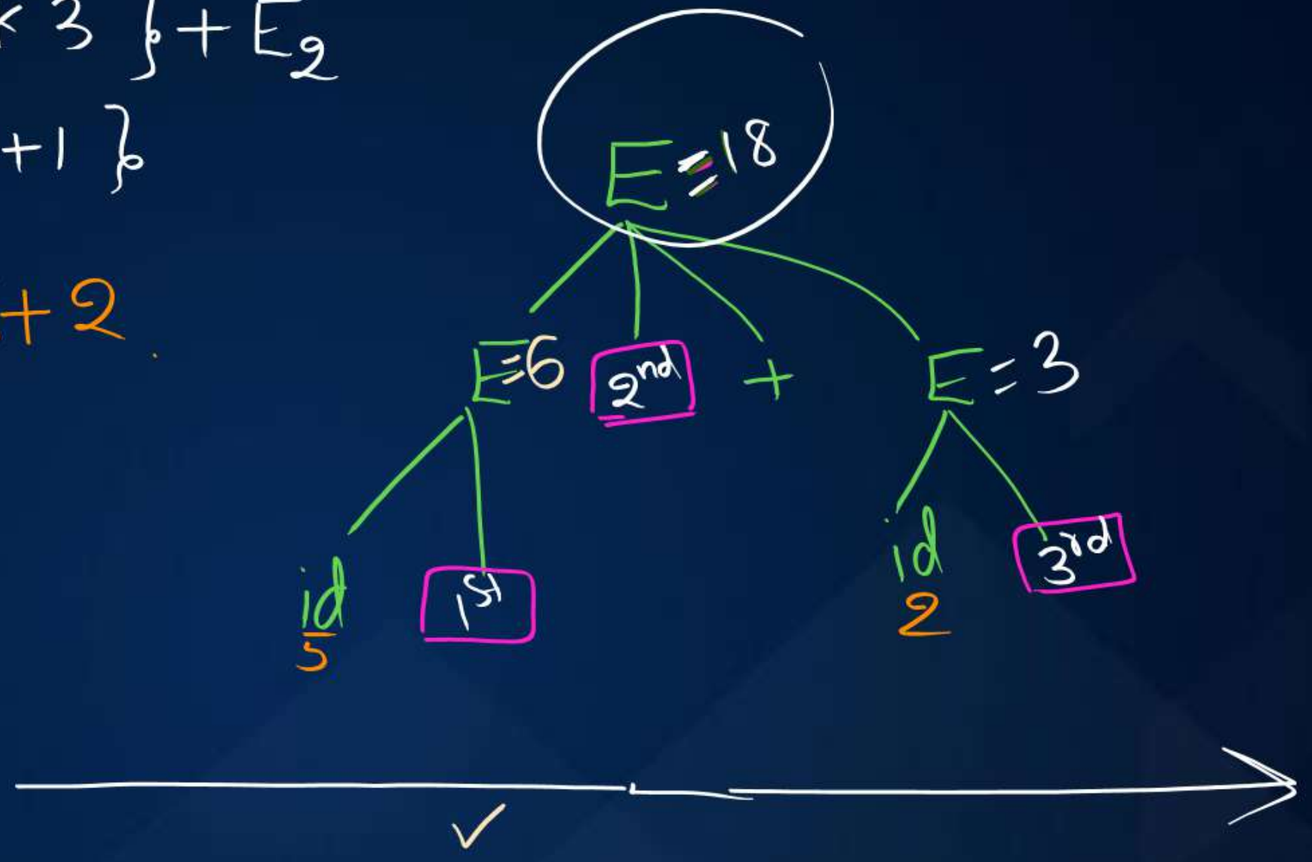
We can't translate.



④ $E \rightarrow E_1 \{ E.x = E_1.x * 3 \} + E_2$

$E \rightarrow id \{ E.x = id.val + 1 \}$

Find translation for 5+2.



$= 18$

(5)

$E \rightarrow E + E \quad \{ \text{print } + \}$

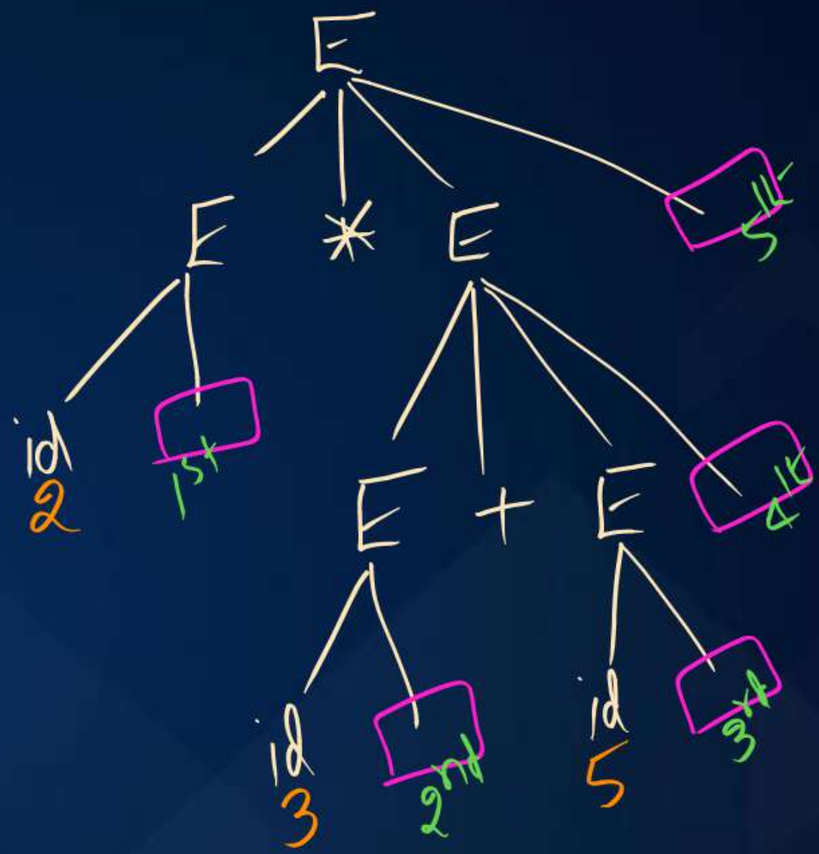
$E \rightarrow E * E \quad \{ \text{print } * \}$

$E \rightarrow \text{id} \quad \{ \text{print id.val} \}$

+ is High precedence

+ and * are Left Associative

Find o/p for 2 * 3 + 5



o/p: 235+*

⑥

$E \rightarrow \{print +\} E + T$

$E \rightarrow id \{print\ id.val\}$

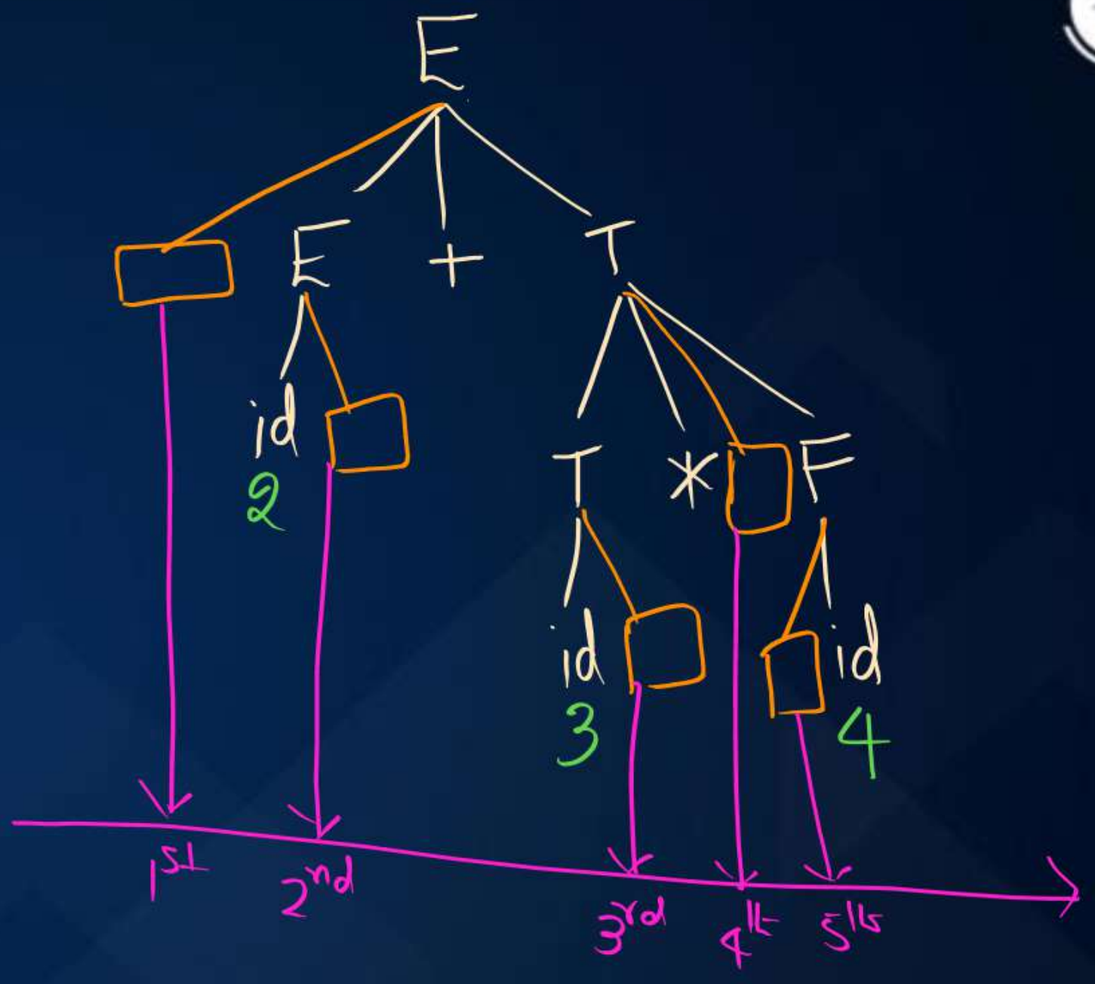
$T \rightarrow T * \{print *\} F$

$T \rightarrow id \{print\ id.val\}$

$F \rightarrow \{print\ id.val\}\ id$

Q1) Find Translation (O/P) for 2+3*4

→ Given SDT is L-attributed



Note: O/p depends on translation position

O/p: +23*4

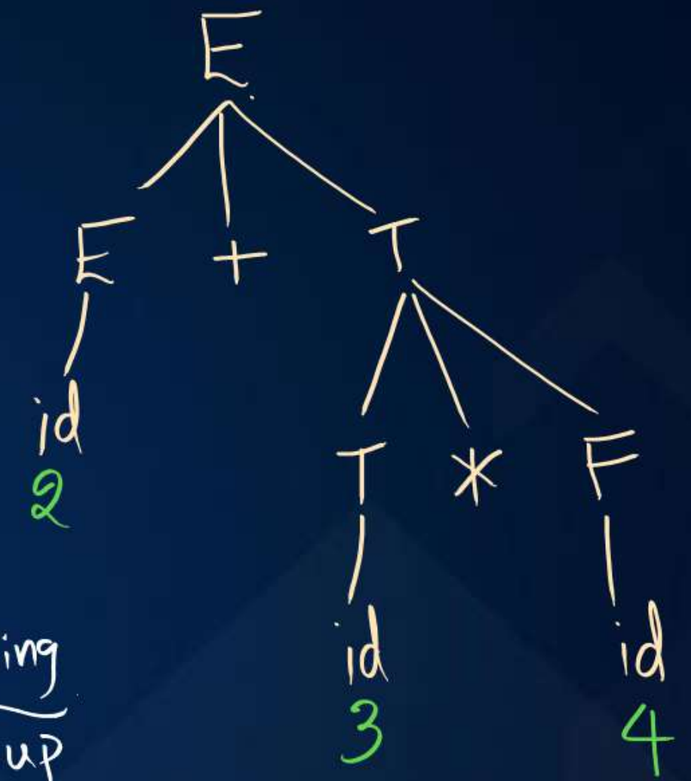
⑥ $E \rightarrow \{print +\} E + T$

$E \rightarrow id \{print\ id.val\}$

$T \rightarrow T * \{print *\} F$

$T \rightarrow id \{print\ id.val\}$

$F \rightarrow \{print\ id.val\}\ id$



Q2) Find Translation (O/P) for 2+3*4 using LR Parsing

Bottom-up

Reverse of RMD

Reverse of RMD : 2 3 4 * +

It is just reverse of RMD O/P

⑥

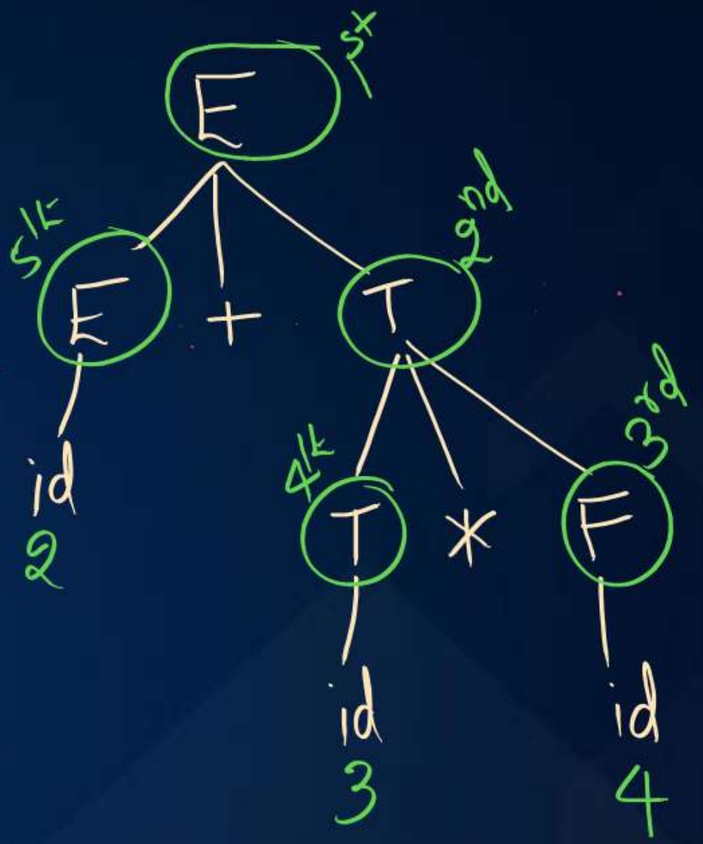
$E \rightarrow \{print +\} E + T$

$E \rightarrow id \{print\ id.val\}$

$T \rightarrow T * \{print *\} F$

$T \rightarrow id \{print\ id.val\}$

$F \rightarrow \{print\ id.val\}\ id$



Q3) Find Translation (O/P) for 2+3*4 using RMD.

RMD order for Non-terminals: 1st 2nd 3rd 4th 5th

Note: O/P only depends on Non-terminal order + * 4 3 2

O/P not depends on translation Position

⑥ $E \rightarrow \{print +\} E + T$

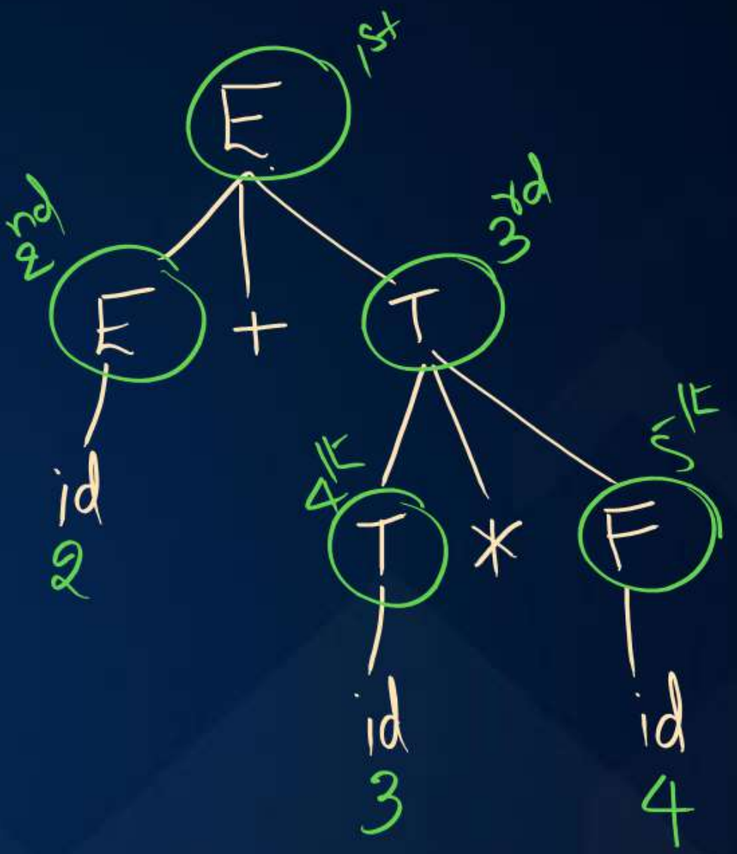
$E \rightarrow id \{print\ id.val\}$

$T \rightarrow T * \{print *\} F$

$T \rightarrow id \{print\ id.val\}$

$F \rightarrow \{print\ id.val\}\ id$

Q4) Find Translation for 2+3*4 using TDP (LMD) (LL(1))
(O/P)



LMD numbering order: 1st 2nd 3rd 4th 5th

+ 2 * 3 4

⑥ $E \rightarrow \{print +\} E + T$

$E \rightarrow id \{print\ id.val\}$

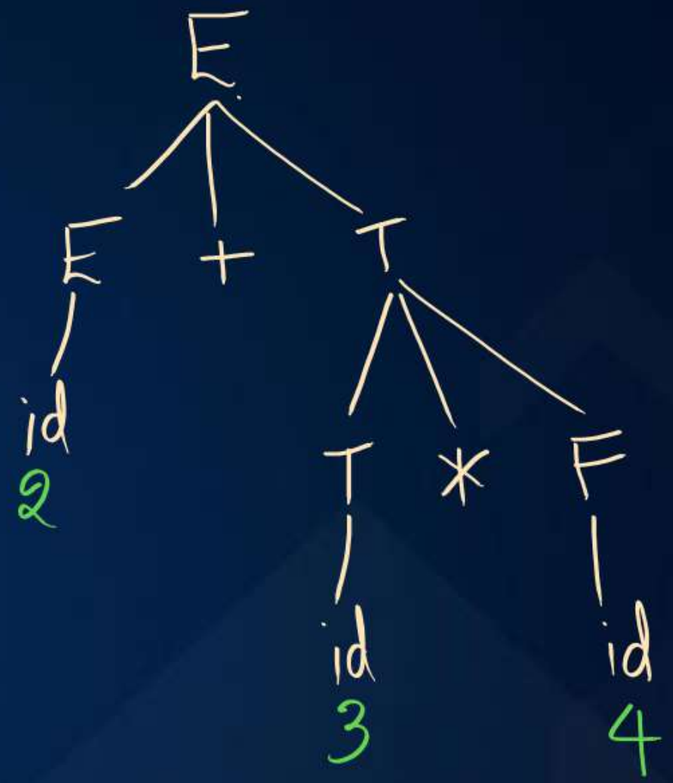
$T \rightarrow T * \{print *\} F$

$T \rightarrow id \{print\ id.val\}$

$F \rightarrow \{print\ id.val\}\ id$

Qs) Find Translation (O/P) for 2+3*4

Using Reverse of LMD.



O/p: 43*2+

Reverse of LMD O/p

⑥ $E \rightarrow \{print +\} E + T$

$E \rightarrow id \{print\ id.val\}$

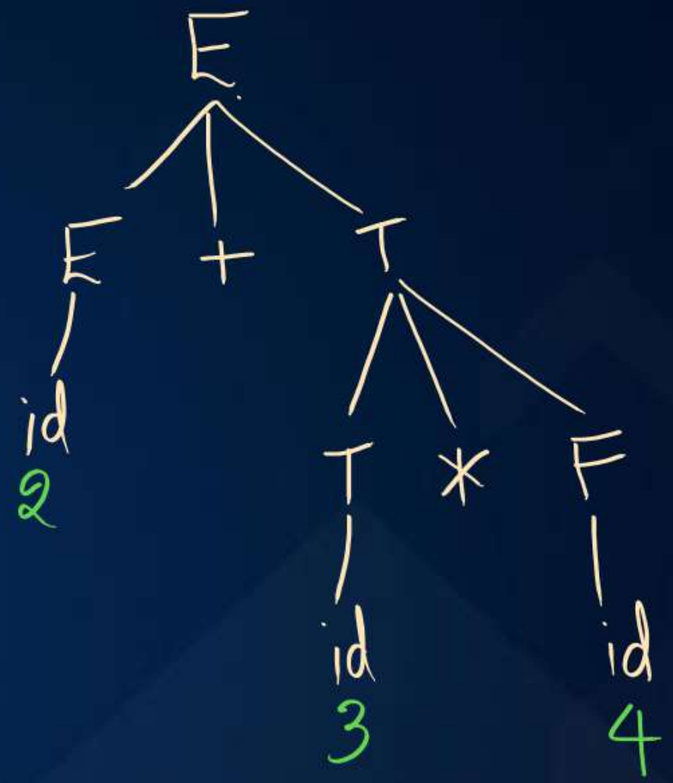
$T \rightarrow T * \{print *\} F$

$T \rightarrow id \{print\ id.val\}$

$F \rightarrow \{print\ id.val\}\ id$

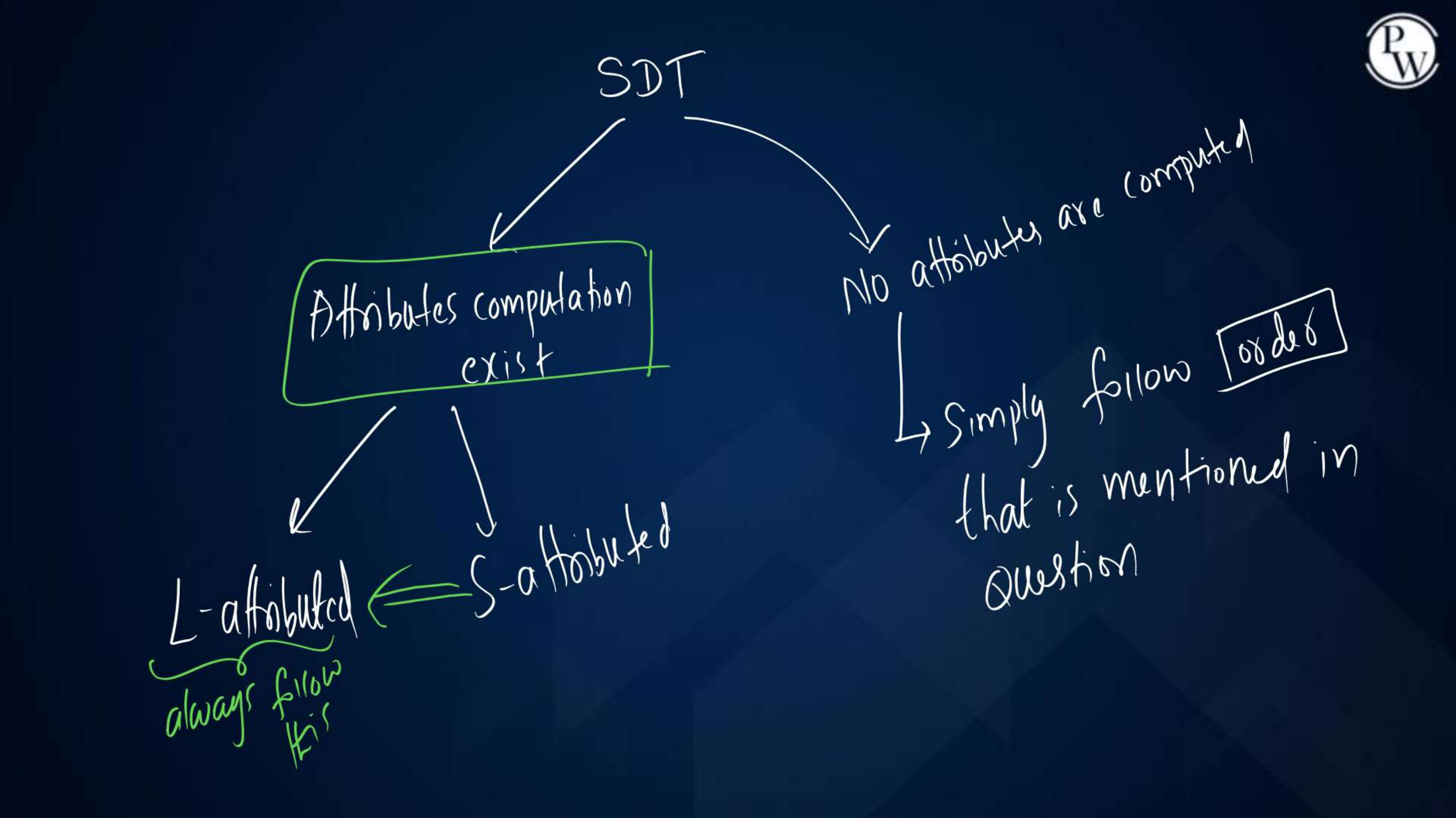
Q(c) Find Translation (O/P) for 2+3*4

Using Reverse of L-attributed order.



O/p: 4*32+

Reverse of L-attributed evaluation

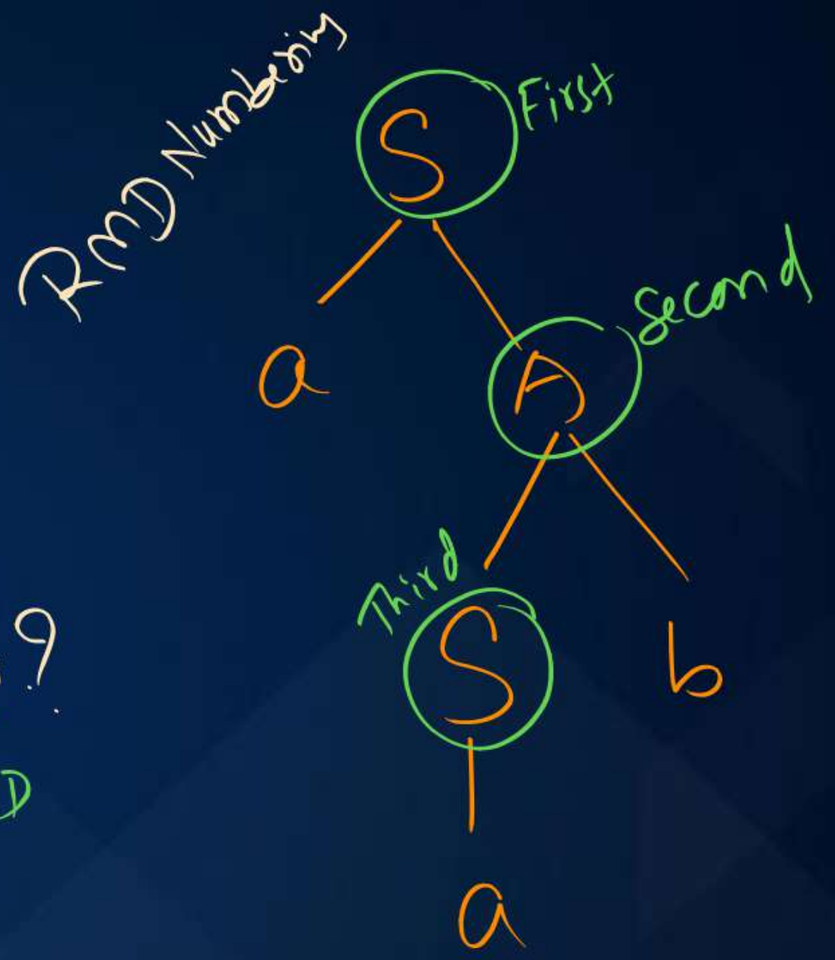

SDT
Attributes computation exist
No attributes are computed
Simply follow order that is mentioned in Question
L-attributed
S-attributed
always follow this

(7)

$S \rightarrow aA$ {point 1}

$S \rightarrow a$ {point 2}

$A \rightarrow Sb$ {point 3}



I/p: aab

what is o/p using Bottom-up parser?

Reverse of RMD

Reverse of RMD: Third Second First

2 3 1

(8)

$E \rightarrow E_1 + E_2 \quad \{E.x = E_1.x + E_2.x + 1\}$

$E \rightarrow E_1 * E_2 \quad \{E.x = E_1.x + E_2.x + 1\}$

$E \rightarrow id \quad \{E.x = 0\}$

What is Functionality of SDT ?

↳ It counts no of operators

(9)

$E \rightarrow E_1 + E_2 \quad \{ E.x = E_1.x + E_2.x \}$

$E \rightarrow E_1 * E_2 \quad \{ E.x = E_1.x + E_2.x \}$

$E \rightarrow id \quad \{ E.x = 1 \}$

What is Functionality of SDT ?

↳ It counts no of operands

10

$E \rightarrow E_1 + E_2 \quad \{ E.x = E_1.x + E_2.x + 1 \}$

$E \rightarrow E_1 * E_2 \quad \{ E.x = E_1.x + E_2.x + 1 \}$

$E \rightarrow id \quad \{ E.x = 1 \}$

What is Functionality of SDT ?

↳ It counts no. of operators & operands


THANK YOU
GW
SOLDIERS !