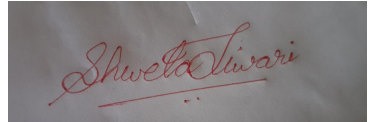


CD: COMPILER DESIGN  
CD: UNIT-3 09/2022

SEPTEMBER 2022 / IT-3rd year, Vth semester  
FALL SEMESTER, YEAR (V/VI, 3rd)  
FALL SESSION (2022-23)  
(CD)  
MS. SHWETA TIWARI  
Published: SEPTEMBER, 2022

PREPARED FOR  
Engineering Students  
All Engineering College



# TOPIC On : UNIT-3

# SYNTAX-DIRECTED DEFINITION

---

By SHWETA TIWARI  
**Under On: Syntax-Directed Translation**

## TOPIC On : UNIT-3

# SYNTAX-DIRECTED DEFINITION

### **1. Syntax Directed Definition (SDD)**

#### ❖ **SDD= CFG+ Semantic Rule**

- A SDD is a context free grammar together with semantic rules.
- Attributes are associated with the grammar symbols and semantic rules are associated with the production.
- If 'x' is a symbol and 'a' is one of its attributes then x.a denotes value at node 'x'.
- y, b then y.b
- Attribute maybe numbers, string, references, data types, etc, .

PRODUCTS	SEMANTIC RULE
$E \rightarrow E + T$	$E.val = E.val + T.val$
$E \rightarrow T$	$E.val = T.val$

### **2. Type of Attributes**

**1. Synthesize Attribute-** If a node takes value from its children then it is called synthesize attribute.

Example:

$A \rightarrow BCD$  (A be a parent node. BCD are children nodes, S is an attribute.)

$A.S \rightarrow B.S$

$A.S \rightarrow C.S$

$A.S \rightarrow D.S$

**2. Inherited Attribute-** If a node takes value from its parent or shivling then it is called inherited attribute.

Example:

$A \rightarrow BCD$  (A be a parent node. BCD are children nodes.)

$C.I \rightarrow A.I$  --- (Patent node) ( I is an attribute.)

$C.I \rightarrow B.I$  ---- (Sibling node)

$C.I \rightarrow D.I$  ---- (Sibling node)

### 3. Type of Syntax Directed Definition (SDD)

1. S-Attributed SDD/S- Attributed Definition/S-Attributed Grammar
2. L-Attributed SDD/L- Attributed Definition/L-Attributed Grammar

Sr.no	S-Attributed SDD	L-Attributed SDD
1.	A SDD that uses only synthesized attributes is called S- Attribute.	A SDD that uses <u>both synthesized and inherited</u> is called L- Attribute SDD but each inherited attribute it's restricted to <u>inherit from parent or left siblings</u> only.
2.	$A \rightarrow BCD$ $A.S \rightarrow B.S$ $A.S \rightarrow C.S$ $A.S \rightarrow D.S$	$A \rightarrow xyz$ $\{y.S = A.S, y.S = x.S, \underline{y.S = z.S}\}$
3.	Semantic actions are always placed at the right and of the production. It's also called postfix SDD.	Semantic actions are placed anywhere on RHS.
4.	Attributes can be evaluated with the bottom of parsing.	Attributes evaluated by traversing parse tree depth first, left to right order that means top down left to right order.

4. Annotated Parse Tree – The parse tree containing the values of attributes at each node for a given input string is called an annotated or decorated parse tree.

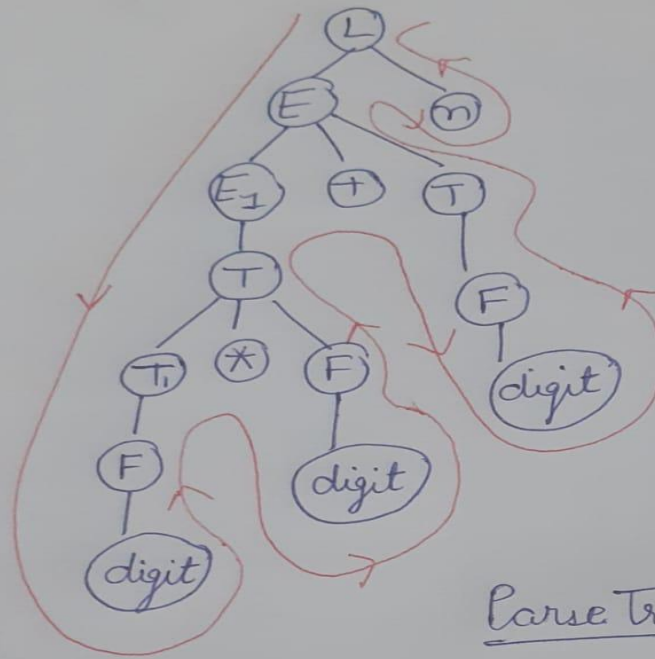
❖ **Features** –

- High level specification
- Hides implementation details
- Explicit order of evaluation is not specified

### 5. SDD of a simple desk calculator / SDD for evaluation of expression/Annotated parse tree for (3\*5+4)

PRODUCTIONS	SEMANTIC RULE
$L \rightarrow E_n$	$L.val = E.val$
$E \rightarrow E_1 + T$	$E.val = E_1.val + T.val$
$E \rightarrow T$	$E.val = T.val$
$T \rightarrow T_1 * F$	$T.val = T_1.val \times F.val$
$T \rightarrow F$	$T.val = F.val$
$F \rightarrow \text{digit}$	$F.val = \text{digit.lexval}$

3 \* 5 + 4



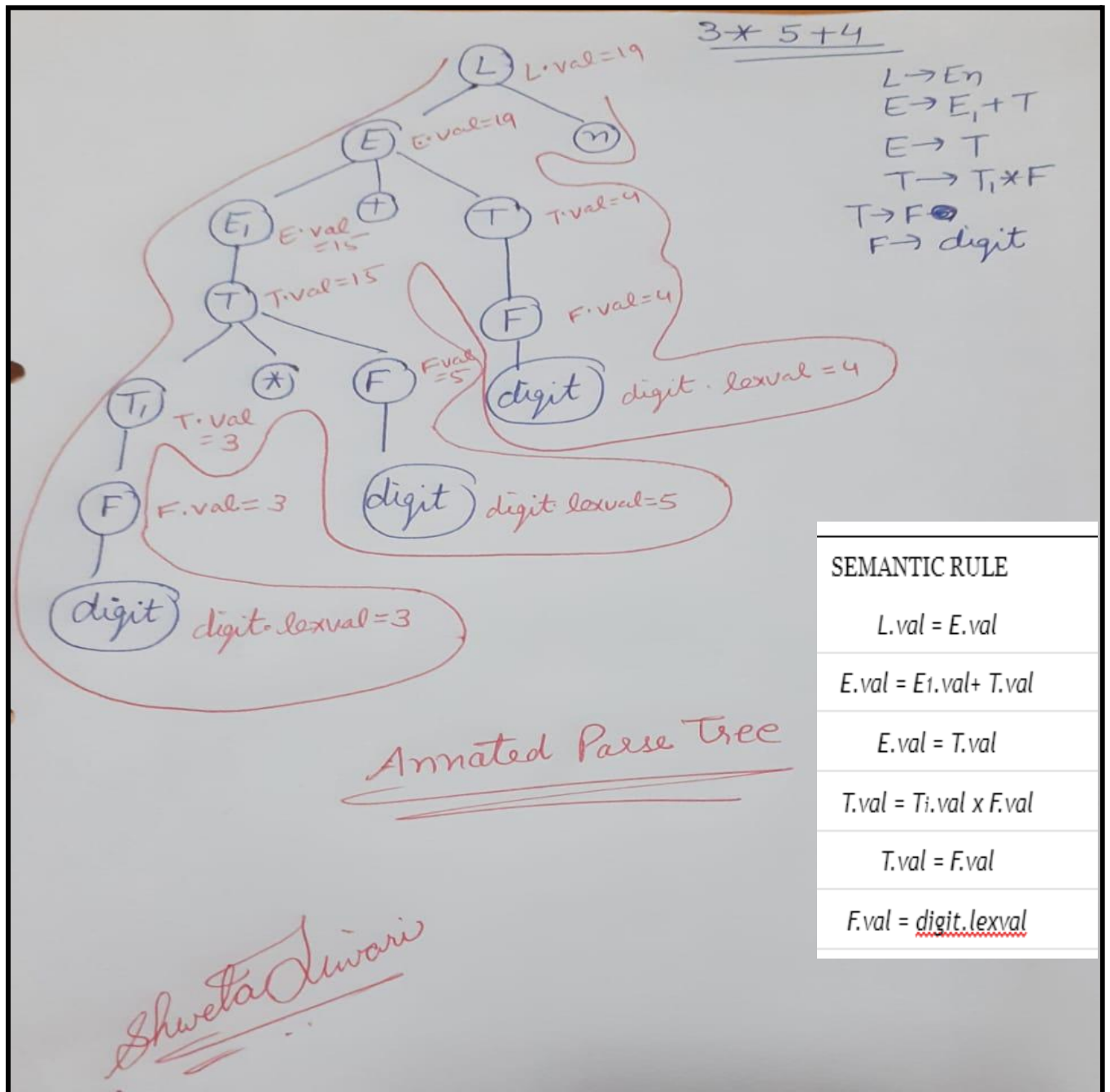
Parse Tree

$L \rightarrow E n$   
 $E \rightarrow E + T$   
 $E \rightarrow T$   
 $T \rightarrow T_1 * F$   
 $T \rightarrow F$   
 $F \rightarrow \text{digit}$   
Production

SEMANTIC RULE
$L.val = E.val$
$E.val = E1.val + T.val$
$E.val = T.val$
$T.val = T1.val * F.val$
$T.val = F.val$
$F.val = \text{digit.lexval}$

Shweta Jivari

**Parse Tree**



Annotated Parse Tree