

COMPILER DESIGN

Syntax Directed Translation

TABLE OF CONTENTS

- SDD

- TYPES OF ATTRIBUTES

- ANNOTATED PARSE TREE

- DEPENDENCY GRAPH

- SDT

- TYPES OF SDT

SDD- Syntax Directed Definitions

□ SDD=CFG(Productions) + SEMANTIC RULES.

Productions

$$E \rightarrow E + T$$

$$T \rightarrow T * F$$

Semantic Rules

$$E.a = E.a + T.a$$

$$T.a = T.a * F.a$$

□ Where E,T and F are grammar symbols or nodes.

□ Left side productions and right side semantic rules.

□ Attributes are associated with grammar symbols.

□ a is the attribute and E.a is the value at node E.

TYPES OF ATTRIBUTES

□ Synthesized Attributes

If a node takes values from its children

eg: $P \rightarrow XYZ$, P – Parent node, XYZ – Children nodes.

$$P.s = X.s$$

$$P.s = Y.s$$

$$P.s = Z.s$$

TYPES OF ATTRIBUTES

□ Inherited Attribute

If a node takes values from either parent node or siblings node.

eg: $P \rightarrow XYZ$

$Y.s = P.s$, parent node

$Y.s = X.s$, left sibling

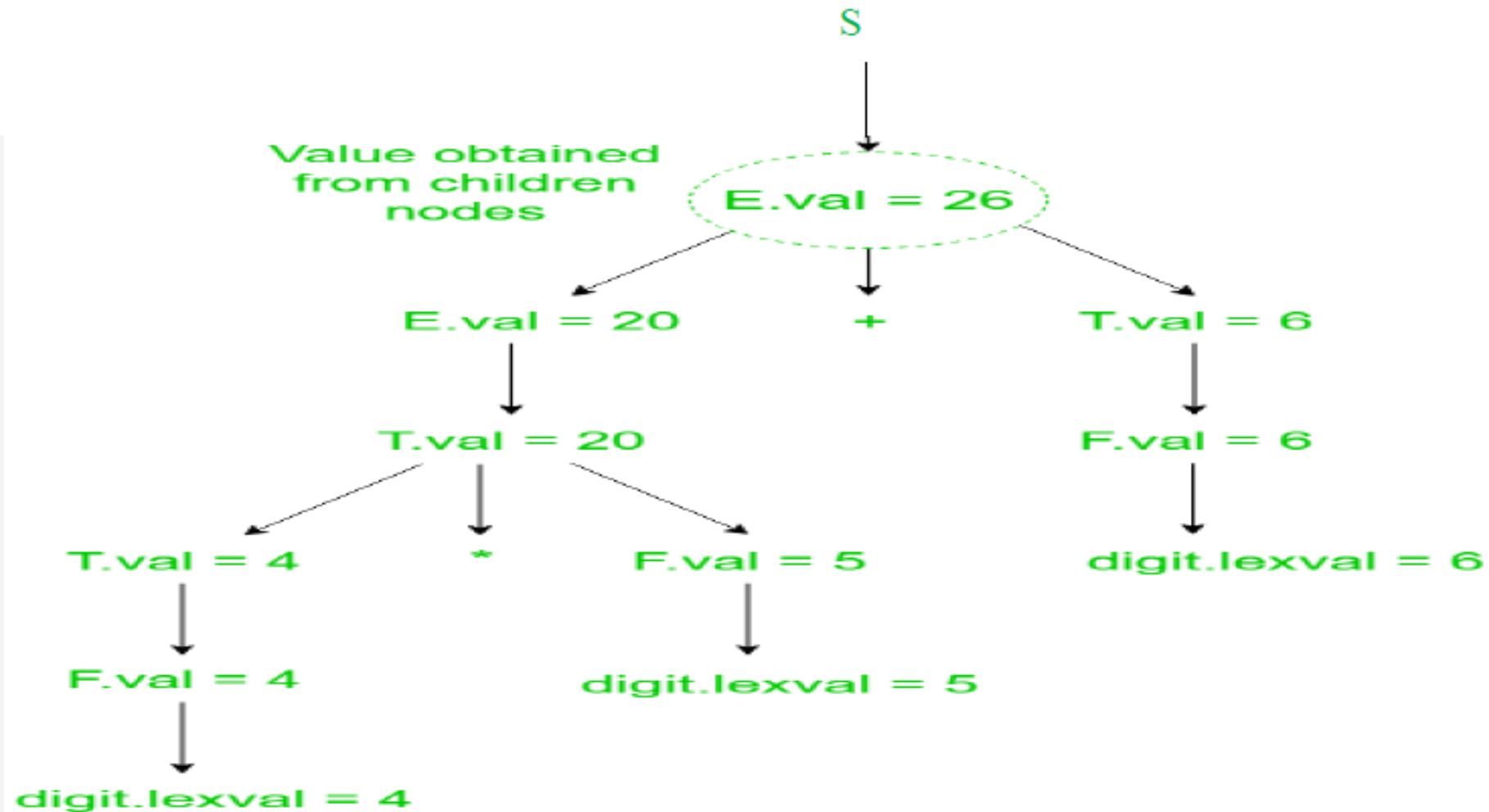
$Y.s = Z.s$, Right sibling

ANNOTATED PARSE TREE

- ❑ A parse tree which shows values at each nodes.
- ❑ Decorated parse tree.

Productions	Semantic Rules
$S \rightarrow E$	$\text{Print}(E.val)$
$E \rightarrow E_1 + T$	$E.val = E_1.val + T.val$
$E \rightarrow T$	$E.val = T.val$
$T \rightarrow T_1 * F$	$T.val = T_1.val * F.val$
$T \rightarrow F$	$T.val = F.val$
$F \rightarrow \text{digit}$	$F.val = \text{digit.lexval}$

ANNOTATED PARSE TREE



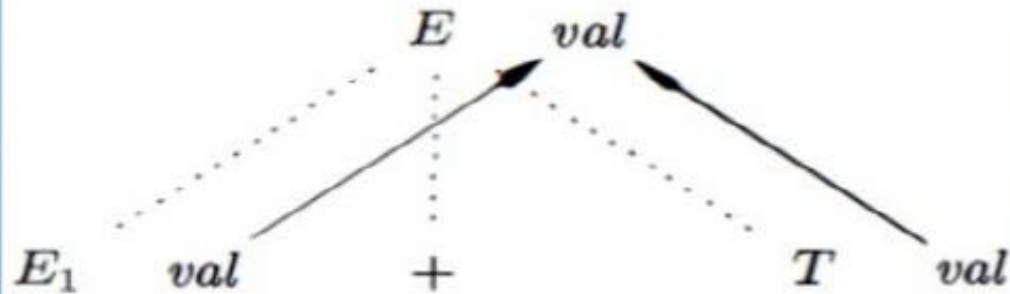
DEPENDENCY GRAPH

PRODUCTION

$E \rightarrow E_1 + T$

SEMANTIC RULE

$E.val = E_1.val + T.val$



DEPENDENCY GRAPH

- ❑ Flow of information among the attributes in a parse tree.
- ❑ Order of evaluation for attributes in a parse tree.
- ❑ Annotated parse tree shows the value of attributes.
- ❑ Dependency graph helps to determine how those values can be computed.

SYNTAX DIRECTED TRANSLATION - SDT

□ SDT = CFG(Productions) + SEMANTIC RULES

Example

$E \rightarrow XY$

$E \rightarrow \{\}XY$

$E \rightarrow X\{\}Y$

$E \rightarrow XY\{\}$

□ semantic Action – in which order the expression will be evaluated

Example $4+5*3$

TYPES OF SDT

S attributed SDT

- 1) Uses only synthesized attributes
 $A \rightarrow XYZ$

- 2) Semantic rules placed at the right end of production
 $E \rightarrow E+T \{E.val = E.val + T.val\}$

- 3) Evaluation of S-attributed SDD, we can use bottom up order of parse tree node

L attributed SDT

- 1) It uses both synthesized & inherited attribute but the production placed at left of the attribute

eg. $A \rightarrow XYZ \{ Y.val = X.val, Z.val = Y.val, X.val = A.val, Y.val = A.val \}$

- 2) It is placed anywhere in the rule
 $A \rightarrow X \{ YZ$
 $A \rightarrow Z \} Z$

- 3) Evaluation of L-attributed SDD, done by depth first & left to right scanning

