

CS & IT Engineering



Compiler Design

Syntax Directed Translations

Lecture : 1



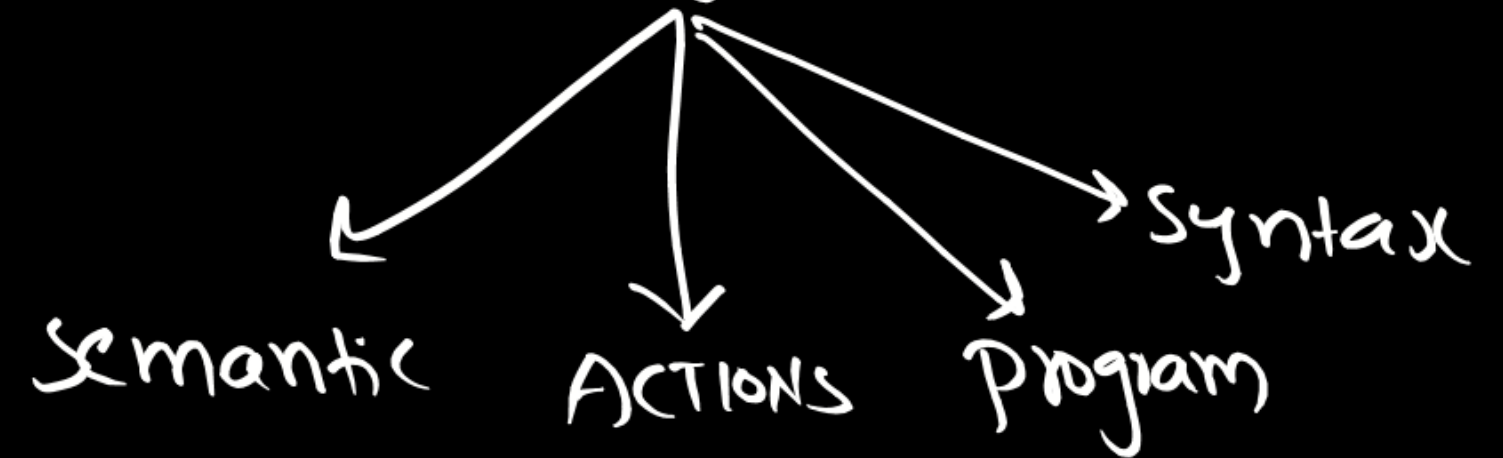
Deva sir

Topics to be covered:

- What is SDT?
- Use of SDT?
- Lexical Vs Syntax Vs Semantic Vs SDT
- Concepts of SDT
 - Attributes of SDT
 - Definitions of SDT
 - Evaluations of SDT

SDD = Syntax + Translation

= CFG + Translation



SDT Applications:



- It can be used to perform
 - i) Semantic analysis
 - ii) Syntax Tree generate
 - iii) Intermediate code generate
 - iv) Generating Parse Tree
 - v) Any meaningful activity
- It can be used to translate expressions
 (Infix/Prefix/postfix → infix/pre/postfix)
- It can be used to translate numbers
 (Binary/Decimal/other → Bin/Decimal/other)
- It can evaluate expressions.

$$E \rightarrow E + E \quad \{ \text{Translation} \}$$
$$E \rightarrow a \quad \{ \text{Translation} \}$$

$\text{SDT} = \text{CFG} + \text{Translation}$

Lexical

Syntax

Semantic

SDT

Tokens

Structures

- Declaration syntax
- if Syntax
- if-else Syntax
- loop Syntax
- Function Syntax
- Expression Syntax

Type checking

- in expression
- in functions
- declare before it use of variable

Anything that
has some logical
information in
Syntax.
More powerful than
compiler

Syntax Directed Translation : (SDT)



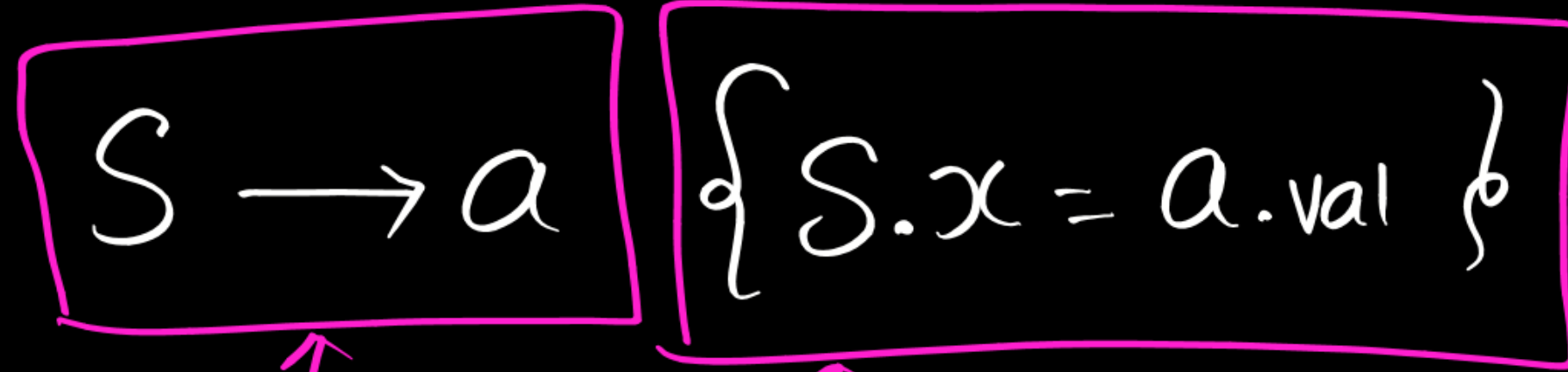
→ 1) Attributes

- Inherited Attribute
- Synthesized Attribute

2) Definitions

- L-attributed SDT
- S-attributed SDT

3) Evaluations



What is CFG?

What is Translation?

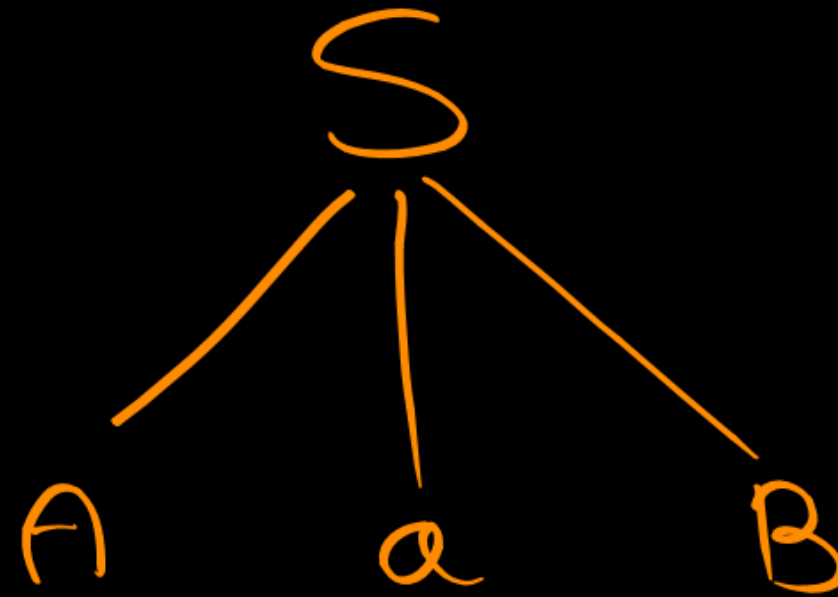
What is Attribute? x , val

What is Definition?

What is Evaluation?

$$S \rightarrow AaB$$

Parent
child
Sibling



production Tree

S: Parent of A, a, B

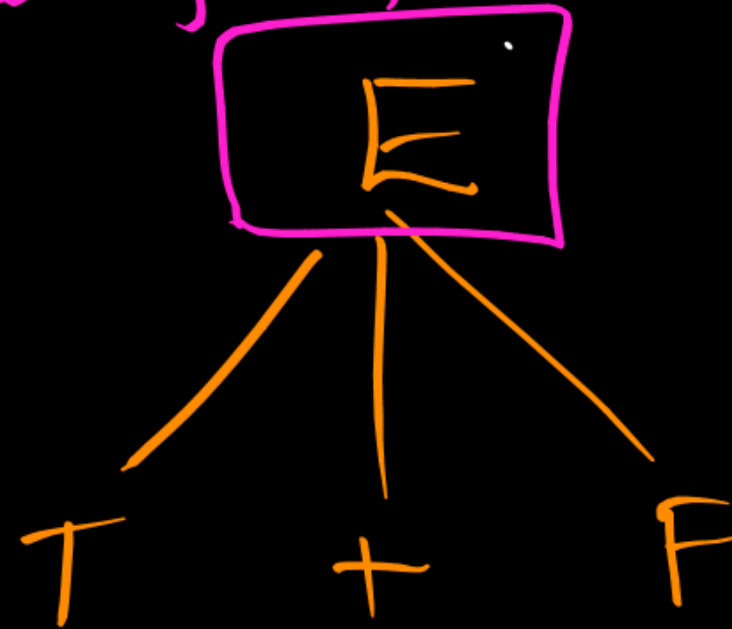
A: child of S, but left Sibling of a and B

a: child of S, Right Sibling of A, left Sibling of B

B: child of S, Right Sibling of A, and a.

$$E \rightarrow T + F \quad \{ E.x = T.y * F.z \}$$

E is computed by depending its children T and F.



Where is ^(Assignment)
^(Definition) computation?

- ☒ A) E
- ☐ B) T
- ☐ C) F
- ☐ D) E, T, and F

Attributes:

① Inherited Attribute :

↳ Computation depends on parent / siblings.
x is inherited attribute

$$S \rightarrow \boxed{A} B \quad \{ \underbrace{A.x = B.y}_{\text{computation}} \}$$

② Synthesized Attribute:

A depends on sibling B

↳ Computation depends on children.

$$\boxed{E} \rightarrow a \quad \{ E.x = a.val \}$$

$$S \rightarrow AB \quad \{ A.x = B.y + S.z \}$$

A
computation

↳ Depends on $\underbrace{B}_{\text{Sibling of A}} \& \underbrace{S}_{\text{Parent of A}}$

$\left. \begin{matrix} x \\ y \\ z \end{matrix} \right\} \text{ attributes}$

x is inherited

Identify the type of attribute.

- ① (i) $E \rightarrow E_1 + E_2 \quad \{ E.x = E_1.x + E_2.x \}$
 depend on children
- (ii) $E \rightarrow id \quad \{ E.x = id.val \}$
 depends on child

Q1) In Rule (i), x is synthesized attribute

Q2) In Rule (ii), x is synthesized attribute

Q3) In SDT, x is synthesized attribute

x } attribute
 val }

② I) $S \rightarrow D \boxed{L}$; $\{ \underline{L.type} = D.type \}$

II) $\boxed{D} \rightarrow \text{int}$ $\{ D.type = \text{int} \}$

III) $L \rightarrow \boxed{L_1}, \text{id}$ $\{ L_1.type = L.type \}$

IV) $L \rightarrow \text{id}$ $\{ \}$

Q1) In Rule I, type is inherited attribute

Q2) In Rule II, type is synthesized attribute

Q3) In Rule III, type is inherited attribute

Q4) In whole SDI, type is neither inherited nor synthesized

$$3) \boxed{S} \rightarrow S_1 S_2 \quad \{ S.\text{count} = S_1.\text{count} + S_2.\text{count} \}$$

$$\boxed{S} \rightarrow (S_1) \quad \{ S.\text{count} = S_1.\text{count} + 1 \}$$

$$S \rightarrow \epsilon \quad \{ \}$$

Count is Synthesized Attribute in above SDT.

4)

$S \rightarrow A a \{ S.x = A.x ; A.y = S.x \}$

$A \rightarrow b \{ A.x = 100 ; A.y = 1000 \}$

- Q1) x is synthesized attribute
- Q2) y is inherited attribute

5

$$E \rightarrow T + F \quad \{ \boxed{E.x} = T.y ; \boxed{F.y} = E.x + 2 ; \boxed{E.y} = F.y - 1 \}$$

$$T \rightarrow \text{id} \quad \{ \boxed{T.x} = 10 ; \boxed{T.y} = \text{id.val} \}$$

$$F \rightarrow \text{id} \quad \{ \boxed{F.x} = \text{id.val} ; \boxed{F.y} = 20 \}$$

Q1) x is synthesized

Q2) y is neither inherited nor synthesized
(Sometimes inherited and sometimes synthesized)

Summary

↳ Types of attributes in SDT ✓

Next: Types of SDTs
(Definitions)

- ↳ L-attributed Grammar
- ↳ S-attributed Grammar

Thank you
PW
Soldiers

