

# Syntax-Directed Translation

①

( Cap. 5 do livro do dragão )

- Método de compilação inteiramente guiado pelo parser
- Os símbolos da gramática passam a ter atributos.
- Os atributos têm um nome e um valor (string, número, ou outra coisa qualquer que queiramos associar ao símbolo.)
- Associamos regras semânticas às produções da gramática.

- Uma regra semântica indica como é que podemos calcular os valores dos atributos associados aos símbolos da produção
- Exemplo:

Produção	Regra Semântica
$E \rightarrow E_1 + T$	$E_{code} = E_1_{code} \parallel T_{code} \parallel '+'$

### Notas

- Na realidade a produção é  $E \rightarrow E + T$
- O subscrito 1 em  $E_1$  serve para distinguir as duas ocorrências do símbolo  $E$ .
- No exemplo, os símbolos  $E$  e  $T$  têm um atributo chamado code.
- O símbolo  $\parallel$  significa concatenação de strings.

• A regra diz-nos que o valor de  $E$ .code é obtido através da concatenação de  $E_1$ .code,  $T$ .code, e o caracter '+'.  
• Em resumo:

- Associamos atributos a símbolos da gramática
- Associamos regras semânticas a produções da gramática

→ estas duas coisas especificam uma "syntax-directed definition" (SDD)

→ Na próxima folha, exemplo retirado do livro.

	PRODUCTION	SEMANTIC RULES
1)	$L \rightarrow E \mathbf{n}$	$L.val = E.val$
2)	$E \rightarrow E_1 + T$	$E.val = E_1.val + T.val$
3)	$E \rightarrow T$	$E.val = T.val$
4)	$T \rightarrow T_1 * F$	$T.val = T_1.val \times F.val$
5)	$T \rightarrow F$	$T.val = F.val$
6)	$F \rightarrow ( E )$	$F.val = E.val$
7)	$F \rightarrow \mathbf{digit}$	$F.val = \mathbf{digit.lexval}$

Figure 5.1: Syntax-directed definition of a simple desk calculator

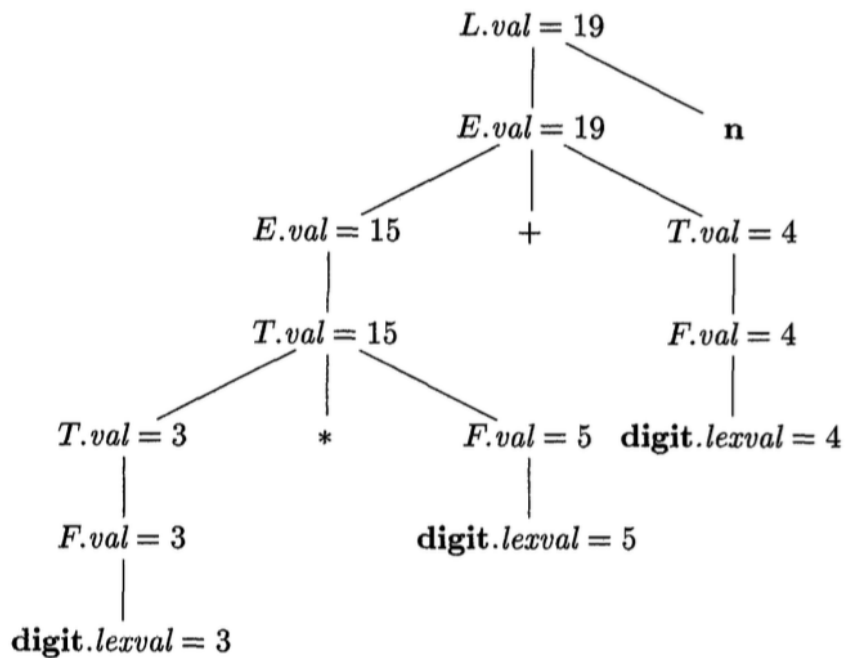


Figure 5.3: Annotated parse tree for  $3 * 5 + 4 \mathbf{n}$

# Outro exemplo.

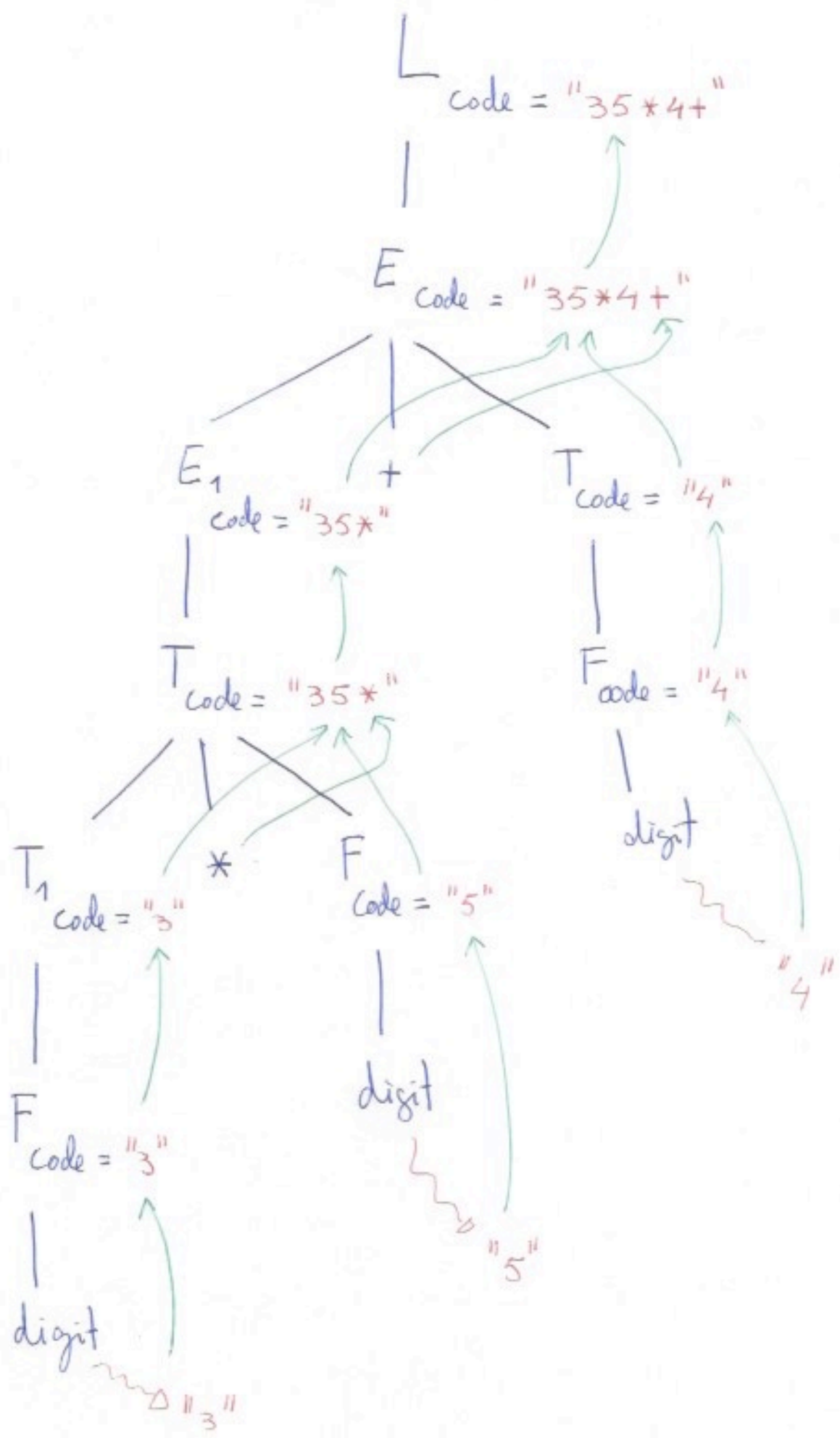
- Vamos supor que queremos converter uma expressão aritmética da notação infixa para a notação posfixa.

Ex:  $3 * 5 + 4 \rightarrow 35 * 4 +$

- Usando a mesma gramática, podemos criar um novo atributo (por exemplo, code) e regras semânticas para fazer a conversão.

Produção	Regra Semântica
$L \rightarrow E$	$L.code = E.code$
$E \rightarrow E_1 + T$	$E.code = E_1.code \parallel T.code \parallel '+'$
$E \rightarrow T$	$E.code = T.code$
$T \rightarrow T_1 * F$	$T.code = T_1.code \parallel F.code \parallel '*'$
$T \rightarrow F$	$T.code = F.code$
$F \rightarrow ( E )$	$F.code = E.code$
$F \rightarrow digit$	$F.code = digit.lexval$

6





- Podemos inclusive calcular os valores dos atributos sem criar explicitamente a árvore.

## Ações semânticas

- Em alternativa (ou em complemento) aos atributos, podemos incluir excertos de código algures no lado direito das produções da gramática.
- Esses excertos de código designam-se por ações semânticas e costumam ser indicadas dentro de chavetas.
- Vejamos novamente o exemplo da tradução de expressões aritméticas infixas  $\rightarrow$  postfixas.





- Estes excertos de código são como se fossem nós artificiais na árvore de parsing.
- Visitar esses nós corresponde a executar os excertos de código em causa.