

# Análise sintática

Função, interação com o compilador  
Análise descendente e ascendente  
Especificação e reconhecimento de cadeias de tokens válidas  
Implementação  
Tratamento de erros

Prof. Thiago A. S. Pardo

1

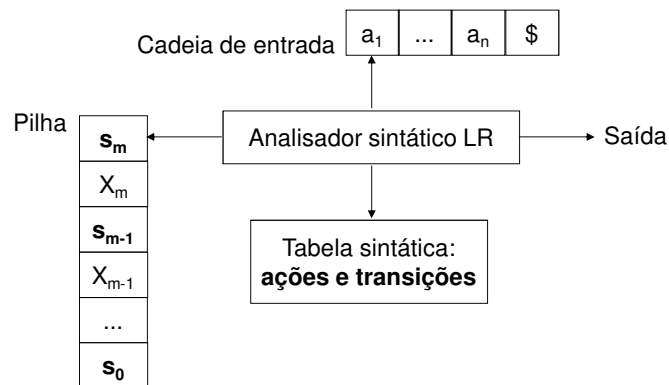
## Análise sintática ascendente

- *Bottom-up*, ascendente ou redutiva
  - Analisadores de precedência de operadores
  - **Analisadores LR**
    - SLR: *Simple LR*
    - LR Canônico
    - *Look Ahead LR*: LALR

2

## Analísadores LR

- Esquema de um analisador LR
  - $X_i$  são símbolos gramaticais
  - $s_i$  são estados que resumizam a informação contida abaixo na pilha



3

## Analísadores LR

- Reconhecer a cadeia  $id*id+id$

- (1)  $\langle E \rangle ::= \langle E \rangle + \langle T \rangle$
- (2)  $\langle E \rangle ::= \langle T \rangle$
- (3)  $\langle T \rangle ::= \langle T \rangle * \langle F \rangle$
- (4)  $\langle T \rangle ::= \langle F \rangle$
- (5)  $\langle F \rangle ::= \langle E \rangle$
- (6)  $\langle F \rangle ::= id$

Tabela sintática LR

Estados	Ações						Transições		
	id	+	*	(	)	\$	E	T	F
0	s5			s4			1	2	3
1		s6				OK			
2		r2	s7		r2	r2			
3		r4	r4		r4	r4			
4	s5			s4			8	2	3
5		r6	r6		r6	r6			
6	s5			s4				9	3
7	s5			s4					10
8		s6			s11				
9		r1	s7		r1	r1			
10		r3	r3		r3	r3			
11		r5	r5		r5	r5			

Na tabela, tem-se que:

- $s_i$  indica "empilhar  $i$ "
- $r_i$  indica "reduzir por regra  $i$ "

De onde vem esse 's'?

## Analísadores LR

- Exercício: reconhecer a cadeia (id)

Pilha	Cadeia	Regra
0	(id)\$	

5

## Analísadores LR

- Exercício: reconhecer a cadeia (id)

Pilha	Cadeia	Regra
0	(id)\$	s4
0(4	id)\$	s5
0(4id5	)\$	r6
0(4F3	)\$	r4
0(4T2	)\$	r2
0(4E8	)\$	s11
0(4E8)11	\$	r5
0F3	\$	r4
0T2	\$	r2
0E1	\$	OK

6

## Analísadores LR

- **Três técnicas** para construir tabelas sintáticas para gramáticas LR
  - **Simple LR (SLR)**
    - Mais fácil de implementar, mas o menos poderoso
  - **Look Ahead LR (LALR)**
    - Complexidade e poder intermediários
  - **LR canônico**
    - Mais complexo, mas mais poderoso
- Tabelas possivelmente distintas para cada técnica, determinando o poder do analisador

7

## Analísadores LR

- **Gramáticas LR**
  - Uma gramática é LR se é possível construir uma tabela sintática LR para ela
    - Não pode haver ambigüidade
- **LL(k) vs. LR(k)**
  - LL(k): decide-se por uma produção olhando-se apenas os k primeiros símbolos da cadeia de entrada
  - LR(k): reconhece-se o lado direito de uma produção tendo visto tudo que foi derivado a partir desse lado direito (na pilha) mais o esquadramento antecipado de k símbolos (da cadeia de entrada)
    - Mais poderoso do que LL(k): pode descrever mais linguagens

8

## Análise SLR

- A análise sintática por meio de uma tabela SLR é chamada **análise sintática SLR**
- Uma gramática é SLR se for possível construir uma tabela SLR para ela

9

## Análise SLR

- A construção da tabela SLR se baseia no **conjunto canônico de itens LR(0)**
  - LR(0): não se olha nenhum símbolo a frente
- Um item para uma gramática G é uma regra de produção com alguma indicação do que já foi derivado/consumido na regra durante a análise sintática
  - Exemplo:  $A \rightarrow XYZ$ 
    - $A \rightarrow .XYZ$
    - $A \rightarrow X.YZ$
    - $A \rightarrow XY.Z$
    - $A \rightarrow XYZ.$
  - Regras do tipo  $A \rightarrow \lambda$  geram somente um item  $A \rightarrow .$

10

## Análise SLR

### ■ Construção do conjunto canônico de itens

#### □ Duas operações

1. Acrescentar à gramática a produção  $S' \rightarrow S$  (em que S é o símbolo inicial da gramática)
  - Permite a identificação do fim da análise, mais especificamente,  $S' \rightarrow S$ .
2. Computar as funções *fechamento* e *transição* para a nova gramática

11

## Análise SLR

### ■ Função *fechamento*

- Seja I um conjunto de itens LR(0)
  1. Todo item em I pertence ao fechamento(I)
  2. Se  $A \rightarrow \alpha.X\beta$  está em fechamento(I) e  $X \rightarrow \gamma$  é uma produção, então adiciona-se  $X \rightarrow .\gamma$  ao conjunto
- Em outras palavras
  - Inicializa-se o conjunto I com as regras iniciais da gramática, colocando-se o indicador (.) no início de cada regra
  - Para cada regra no conjunto, adicionam-se as regras dos não terminais que aparecem precedidos pelo indicador (.)

12

## Análise SLR

- Exemplo

$S' \rightarrow S$

$S \rightarrow a \mid [L]$

$L \rightarrow L;S \mid S$

$I = \{S \rightarrow [.L]\}$

fechamento(I) =

13

## Análise SLR

- Exemplo

$S' \rightarrow S$

$S \rightarrow a \mid [L]$

$L \rightarrow L;S \mid S$

$I = \{S \rightarrow [.L]\}$

fechamento(I) =  $\{S \rightarrow [.L], L \rightarrow .L;S, L \rightarrow .S, S \rightarrow .a, S \rightarrow .[L]\}$

14

## Análise SLR

### ■ Função *transição*

- *transição*(I,X): consiste avançar o indicador (.) através do símbolo gramatical X das produções correspondentes em I e calcular a função *fechamento* para o novo conjunto

### ■ Exemplo

$I = \{S \rightarrow [L., L \rightarrow L.; S\}$

$\text{transição}(I,;) =$

$S' \rightarrow S$   
 $S \rightarrow a \mid [L]$   
 $L \rightarrow L; S \mid S$

## Análise SLR

### ■ Função *transição*

- *transição*(I,X): consiste avançar o indicador (.) através do símbolo gramatical X das produções correspondentes em I e calcular a função *fechamento* para o novo conjunto

### ■ Exemplo

$I = \{S \rightarrow [L., L \rightarrow L.; S\}$

$\text{transição}(I,;) = \{L \rightarrow L.; S, S \rightarrow .a, S \rightarrow .[L]\}$

$S' \rightarrow S$   
 $S \rightarrow a \mid [L]$   
 $L \rightarrow L; S \mid S$



## Análise SLR

- Algoritmo para obter o conjunto canônico de itens LR(0)

$C := \{I_0 = \text{fechamento}(\{S' \rightarrow S\})\}$

repita

para cada conjunto  $I$  em  $C$  e  $X$  símbolo de  $G$ , tal que  
transição  $(I, X) \neq \lambda$

adicione transição  $(I, X)$  a  $C$

até que todos os conjuntos tenham sido adicionados a  $C$

17

## Análise SLR

- Exemplo

0)  $S' \rightarrow S$

1)  $S \rightarrow a$

2)  $S \rightarrow [L]$

3)  $L \rightarrow L;S$

4)  $L \rightarrow S$

18

## Análise SLR

### Exemplo

Conjunto de itens

0) $S' \rightarrow S$	$I_0 = \{S' \rightarrow \cdot S, S \rightarrow \cdot a, S \rightarrow \cdot [L]\}$
1) $S \rightarrow a$	transição( $I_0, S$ ) = $\{S' \rightarrow S \cdot\} = I_1$
2) $S \rightarrow [L]$	transição( $I_0, a$ ) = $\{S \rightarrow a \cdot\} = I_2$
3) $L \rightarrow L;S$	transição( $I_0, [$ ) = $\{S \rightarrow [ \cdot L], L \rightarrow \cdot L;S, L \rightarrow \cdot S, S \rightarrow \cdot a, S \rightarrow \cdot [L]\} = I_3$
4) $L \rightarrow S$	transição( $I_3, L$ ) = $\{S \rightarrow [L \cdot], L \rightarrow L \cdot ;S\} = I_4$
	transição( $I_3, S$ ) = $\{L \rightarrow S \cdot\} = I_5$
	transição( $I_3, a$ ) = $\{S \rightarrow a \cdot\} = I_2$
	transição( $I_3, [$ ) = $\{S \rightarrow [ \cdot L], L \rightarrow \cdot L;S, L \rightarrow \cdot S, S \rightarrow \cdot a, S \rightarrow \cdot [L]\} = I_3$
	transição( $I_4, ]$ ) = $\{S \rightarrow [L] \cdot\} = I_6$
	transição( $I_4, ;$ ) = $\{L \rightarrow L; \cdot S, S \rightarrow \cdot a, S \rightarrow \cdot [L]\} = I_7$
	transição( $I_7, S$ ) = $\{L \rightarrow L;S \cdot\} = I_8$
	transição( $I_7, a$ ) = $\{S \rightarrow a \cdot\} = I_2$
	transição( $I_7, [$ ) = $\{S \rightarrow [ \cdot L], L \rightarrow \cdot L;S, L \rightarrow \cdot S, S \rightarrow \cdot a, S \rightarrow \cdot [L]\} = I_3$

19

## Análise SLR

### Construção da tabela sintática

- Seja  $C = \{I_0, I_1, \dots, I_n\}$ , os estados são  $0 \dots n$ , com 0 sendo o estado inicial
- A linha  $i$  da tabela é construída pelo conjunto  $I_i$ 
  - Ações na tabela
    - Se transição( $I_i, a$ )= $I_j$ , então ação[ $i, a$ ]= $s_j$
    - Com exceção da regra  $S' \rightarrow S$  adicionada, para todas as outras regras, se  $A \rightarrow \alpha$  está em  $I_i$ , então, para todo  $a$  em seguidor( $A$ ), faça ação[ $i, a$ ]= $rn$ , em que  $n$  é o número da produção  $A \rightarrow \alpha$
    - Se  $S' \rightarrow S$  está em  $I_i$ , então faça ação[ $i, \$$ ]=OK
  - Transições na tabela
    - Se transição( $I_i, A$ )= $I_j$ , então transição( $i, A$ )= $j$

- Entradas não definidas indicam erros
- Ações conflitantes indicam que a gramática não é SLR

20

## Análise SLR

- Construção da tabela sintática

- 0)  $S' \rightarrow S$
- 1)  $S \rightarrow a$
- 2)  $S \rightarrow [L]$
- 3)  $L \rightarrow L;S$
- 4)  $L \rightarrow S$

$S(S') = \{\$ \}$   
 $S(S) = S(S') \cup S(L) = \{\$, ., ; \}$   
 $S(L) = \{[, ; \}$

OBS: agora, para indicar fim de cadeia, utiliza-se o \$  
Antes estávamos sobrecarregando o símbolo de cadeia vazia  $\lambda$

21

## Análise SLR

- Construção da tabela sintática

- 0)  $S' \rightarrow S$
- 1)  $S \rightarrow a$
- 2)  $S \rightarrow [L]$
- 3)  $L \rightarrow L;S$
- 4)  $L \rightarrow S$

$S(S') = \{\$ \}$   
 $S(S) = S(S') \cup S(L) = \{\$, ., ; \}$   
 $S(L) = \{[, ; \}$

Tabela sintática SLR

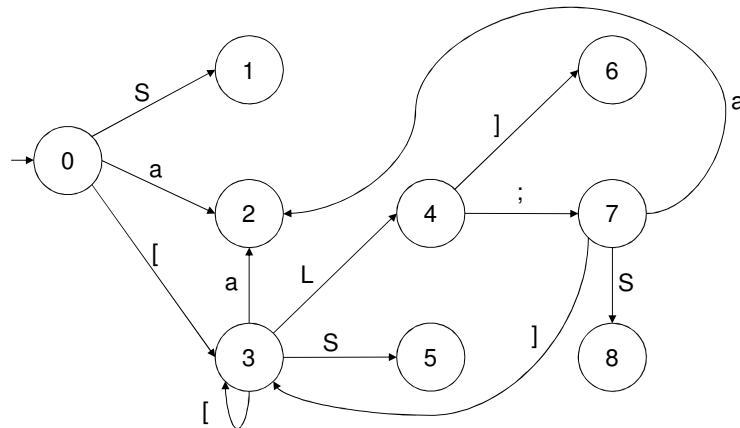
Estados	Ações					Transições	
	a	[	]	;	\$	S	L
0	s2	s3				1	
1					OK		
2			r1	r1	r1		
3	s2	s3				5	4
4			s6	s7			
5			r4	r4			
6			r2	r2	r2		
7	s2	s3				8	
8			r3	r3			

OBS: agora, para indicar fim de cadeia, utiliza-se o \$  
Antes estávamos sobrecarregando o símbolo de cadeia vazia  $\lambda$

22

## Análise SLR

- Autômato correspondente



23

## Análise SLR

- O autômato não é usado para reconhecer cadeias, mas para acompanhar o estado da análise sintática
- O analisador aceitará uma cadeia quando ocorrer uma redução pela regra adicionada  $S' \rightarrow S$

24

## Análise SLR

- **Exercício em duplas:** construir a tabela sintática para a gramática abaixo

$S \rightarrow \text{if } E \text{ then } C \mid C$

$E \rightarrow a$

$C \rightarrow b$

25

## Análise SLR

- Passo 1: adicionar a regra  $S' \rightarrow S$

0)  $S' \rightarrow S$

1)  $S \rightarrow \text{if } E \text{ then } C$

2)  $S \rightarrow C$

3)  $E \rightarrow a$

4)  $C \rightarrow b$

26

## Análise SLR

- Passo 2: construir o conjunto de itens

- 0)  $S' \rightarrow S$
- 1)  $S \rightarrow \text{if } E \text{ then } C$
- 2)  $S \rightarrow C$
- 3)  $E \rightarrow a$
- 4)  $C \rightarrow b$

27

## Análise SLR

- Passo 2: construir o conjunto de itens

- 0)  $S' \rightarrow S$
- 1)  $S \rightarrow \text{if } E \text{ then } C$
- 2)  $S \rightarrow C$
- 3)  $E \rightarrow a$
- 4)  $C \rightarrow b$

Conjunto de itens

$$\begin{aligned} I_0 &= \{S' \rightarrow .S, S \rightarrow .\text{if } E \text{ then } C, S \rightarrow .C, C \rightarrow .b\} \\ t(I_0, S) &= \{S' \rightarrow S.\} = I_1 \\ t(I_0, \text{if}) &= \{S \rightarrow \text{if } .E \text{ then } C, E \rightarrow .a\} = I_2 \\ t(I_0, C) &= \{S \rightarrow C.\} = I_3 \\ t(I_0, b) &= \{C \rightarrow b.\} = I_4 \\ t(I_2, E) &= \{S \rightarrow \text{if } E .\text{then } C\} = I_5 \\ t(I_2, a) &= \{E \rightarrow a.\} = I_6 \\ t(I_5, \text{then}) &= \{S \rightarrow \text{if } E \text{ then } .C, C \rightarrow .b\} = I_7 \\ t(I_7, C) &= \{S \rightarrow \text{if } E \text{ then } C.\} = I_8 \\ t(I_7, b) &= \{C \rightarrow b.\} = I_4 \end{aligned}$$

28

## Análise SLR

- Passo 3: construir a tabela sintática

- 0)  $S' \rightarrow S$
- 1)  $S \rightarrow \text{if } E \text{ then } C$
- 2)  $S \rightarrow C$
- 3)  $E \rightarrow a$
- 4)  $C \rightarrow b$

$S(S') = \{\$ \}$   
 $S(S) = S(S') = \{\$ \}$   
 $S(E) = \{\text{then}\}$   
 $S(C) = S(S) = \{\$ \}$

29

## Análise SLR

- Passo 3: construir a tabela sintática

- 0)  $S' \rightarrow S$
- 1)  $S \rightarrow \text{if } E \text{ then } C$
- 2)  $S \rightarrow C$
- 3)  $E \rightarrow a$
- 4)  $C \rightarrow b$

$S(S') = \{\$ \}$   
 $S(S) = S(S') = \{\$ \}$   
 $S(E) = \{\text{then}\}$   
 $S(C) = S(S) = \{\$ \}$

Tabela sintática SLR

Estados	Ações					Transições		
	if	then	a	b	\$	S	E	C
0	s2			s4		1		3
1					OK			
2			s6				5	
3					r2			
4					r4			
5		s7						
6		r3						
7				s4				8
8					r1			

30

## Análise SLR

- Exercício: reconhecer a cadeia *if a then b*

Pilha	Cadeia	Regra
0	if a then b \$	

31

## Análise SLR

- Exercício: reconhecer a cadeia *if a then b*

Pilha	Cadeia	Regra
0	if a then b \$	s2
0 if 2	a then b \$	s6
0 if 2 a 6	then b \$	r3
0 if 2 E 5	then b \$	s7
0 if 2 E 5 then 7	b \$	s4
0 if 2 E 5 then 7 b 4	\$	r4
0 if 2 E 5 then 7 C 8	\$	r1
0 S 1	\$	OK

32