

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

- Parte-se dos **símbolos terminais em direção ao símbolo inicial** da gramática
- Derivação mais à direita

2

Análise sintática ascendente

- **Redução**: operação de substituição do lado direito de uma produção pelo não-terminal correspondente do lado esquerdo
 - Para a regra $A \rightarrow \alpha$, α pode ser reduzido em A
- Analisadores sintáticos ascendentes
 - Analisadores de **empilha-reduz** (*shift-reduce*)

3

Análise sintática ascendente

- **Componentes** do analisador ascendente
 - **Pilha**, onde os símbolos a serem reduzidos são empilhados
 - **Tabela sintática** que guia o processo de empilhamento e redução
- **Processo** de reconhecimento de uma sentença
 - Empilhar símbolos da cadeia de entrada
 - Quando um lado direito apropriado de uma produção aparece, ele é reduzido (substituído) pelo lado esquerdo da produção
 - Se a análise tiver sucesso, esse processo ocorre até que os símbolos da cadeia de entrada sejam todos consumidos e a pilha fique apenas com o símbolo inicial da gramática

4

Análise sintática ascendente

■ Exemplo

$\langle S \rangle ::= [\langle L \rangle] \mid a$
 $\langle L \rangle ::= \langle L \rangle ; \langle S \rangle \mid \langle S \rangle$

Reconhecer a cadeia [a;a]

Pilha	Cadeia	Regra
\$	[a;a]\$	

5

Análise sintática ascendente

■ Exemplo

$\langle S \rangle ::= [\langle L \rangle] \mid a$
 $\langle L \rangle ::= \langle L \rangle ; \langle S \rangle \mid \langle S \rangle$

Reconhecer a cadeia [a;a]

Pilha	Cadeia	Regra
\$	[a;a]\$	empilha [
\$[a;a]\$	empilha a
\$[a	;a]\$	reduz $S \rightarrow a$
\$[S	;a]\$	reduz $L \rightarrow S$
\$[L	;a]\$	empilha ;
\$[L;	a]\$	empilha a
\$[L;a]\$	reduz $S \rightarrow a$
\$[L;S]\$	reduz $L \rightarrow L;S$
\$[L]\$	empilha]
\$[L]	\$	reduz $S \rightarrow [L]$
\$S	\$	SUCESSO

6

Análise sintática ascendente

- O analisador empilha símbolos até ter na pilha uma seqüência de símbolos que corresponde à definição de um não terminal
 - Seqüência de símbolos: lado direito da produção
 - Não terminal: lado esquerdo da produção
- *Handle*
 - Produção cujo lado direito está na pilha
- Operação de **redução**: substituição do **lado direito do handle** pelo seu **lado esquerdo**
 - O uso da **seqüência correta de handles** no processo de análise leva ao símbolo inicial da gramática
 - Derivação mais a direita para a cadeia de entrada

7

Análise sintática ascendente

Exemplo

$\langle S \rangle ::= [\langle L \rangle] \mid a$
 $\langle L \rangle ::= \langle L \rangle ; \langle S \rangle \mid \langle S \rangle$

Haveria outras opções de handles?

Pilha	Cadeia	Regra
\$	[a;a]\$	empilha [
\$[a;a]\$	empilha a
\$[a	;a]\$	reduz $S \rightarrow a$
\$[S	;a]\$	reduz $L \rightarrow S$
\$[L	;a]\$	empilha ;
\$[L;	a]\$	empilha a
\$[L;a]\$	reduz $S \rightarrow a$
\$[L;S]\$	reduz $L \rightarrow L;S$
\$[L]\$	empilha]
\$[L]	\$	reduz $S \rightarrow [L]$
\$S	\$	SUCESSO

8

Análise sintática ascendente

Exemplo

$\langle S \rangle ::= [\langle L \rangle] \mid a$
 $\langle L \rangle ::= \langle L \rangle ; \langle S \rangle \mid \langle S \rangle$

Haveria outras opções de handles?

- $L \rightarrow S$
 - O que aconteceria?

Pilha	Cadeia	Regra
\$	[a;a]\$	empilha [
\$[a;a]\$	empilha a
\$[a	;a]\$	reduz $S \rightarrow a$
\$[S	;a]\$	reduz $L \rightarrow S$
\$[L	;a]\$	empilha ;
\$[L;	a]\$	empilha a
\$[L;a]\$	reduz $S \rightarrow a$
\$[L;S]\$	reduz $L \rightarrow L;S$
\$[L]\$	empilha]
\$[L]	\$	reduz $S \rightarrow [L]$
\$S	\$	SUCESSO

9

Análise sintática ascendente

Operações durante a análise

- **Empilha:** coloca-se no topo da pilha o primeiro símbolo da cadeia de entrada
- **Reduz:** substitui-se a lado direito do *handle* pelo seu lado esquerdo
- **Aceita:** a cadeia de entrada é reconhecida
- **Erro:** a cadeia de entrada não é reconhecida

10

Análise sintática ascendente

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

11

ASA: precedência de operadores

- Simples e eficiente
- Aplicada, principalmente, para o reconhecimento de expressões
- Subclasse de gramáticas
 - Gramáticas de (precedência de) operadores
 - Não há símbolos não terminais adjacentes
 - Não há produções que derivam a cadeia nula

12

ASA: precedência de operadores

- Exemplo: a gramática abaixo não é de precedência de operadores

$$\begin{aligned} \langle E \rangle &::= \langle E \rangle \langle O \rangle \langle E \rangle \mid (\langle E \rangle) \mid \text{id} \\ \langle O \rangle &::= + \mid - \end{aligned}$$

- Transformando-a em gramática de operadores:

$$\langle E \rangle ::= \langle E \rangle + \langle E \rangle \mid \langle E \rangle - \langle E \rangle \mid (\langle E \rangle) \mid \text{id}$$

13

ASA: precedência de operadores

- Para identificar os *handles*, utilizam-se **relações de precedência** existentes entre os símbolos terminais (operandos e operadores) em uma tabela sintática (ou de precedência)

- $<, > \text{ e } =$

- Considere os terminais a e b

- $a < b$ significa que a tem precedência menor do que b
- $a = b$ significa que a e b têm a mesma precedência
- $a > b$ significa que a tem precedência maior do que b

- Durante a análise ascendente, na pilha:

- $<$ identifica o limite esquerdo do lado direito do *handle*
- $=$ indica que os terminais envolvidos pertencem ao mesmo *handle*
- $>$ identifica o limite direito do lado direito do *handle*

14

ASA: precedência de operadores

■ Tabela sintática

- Matriz quadrada que relaciona todos os terminais da gramática e o símbolo delimitador utilizado (\$)
- Primeira linha da tabela: terminais da cadeia sendo analisada
- Primeira coluna da tabela: terminais do topo da pilha

	id	+	*	→ cadeia
id		>	>	
+	<	>	<	
*	<	>	>	

↓
pilha

- Poucos terminais são operadores

Onde estão os não terminais?

15

ASA: precedência de operadores

■ Uso da tabela sintática

- Seja a o terminal mais ao topo da pilha (os não terminais são ignorados) e b o primeiro terminal da cadeia sendo analisada
 - Se $a < b$ ou $a = b$, então se empilha b
 - Se $a > b$, então se procura o lado direito do *handle* na pilha e o substitui pelo seu lado esquerdo
- O lado direito do *handle* estará delimitado na pilha pelos símbolos $<$ e $>$
 - Os não terminais não precisam aparecer, mas se deve saber que foram produzidos e que seus derivados correspondentes foram consumidos

16

ASA: precedência de operadores

Exemplo: expressões lógicas

$\langle E \rangle ::= \langle E \rangle / \langle T \rangle \mid \langle T \rangle$
 $\langle T \rangle ::= \langle T \rangle \& \langle F \rangle \mid \langle F \rangle$
 $\langle F \rangle ::= (\langle E \rangle) \mid \text{id}$

Tabela sintática

	id	/	&	()	\$
id		>	>		>	>
/	<	>	<	<	>	>
&	<	>	>	<	>	>
(<	<	<	<	=	
)		>	>		>	>
\$	<	<	<	<		

Pilha	Cadeia	Regra
\$	id&id/id\$	

17

ASA: precedência de operadores

Exemplo: expressões lógicas

$\langle E \rangle ::= \langle E \rangle / \langle T \rangle \mid \langle T \rangle$
 $\langle T \rangle ::= \langle T \rangle \& \langle F \rangle \mid \langle F \rangle$
 $\langle F \rangle ::= (\langle E \rangle) \mid \text{id}$

Tabela sintática

	id	/	&	()	\$
id		>	>		>	>
/	<	>	<	<	>	>
&	<	>	>	<	>	>
(<	<	<	<	=	
)		>	>		>	>
\$	<	<	<	<		

Pilha	Cadeia	Regra
\$<	id&id/id\$	empilha
\$<id>	&id/id\$	reduz
\$<	&id/id\$	empilha
\$<&<	id/id\$	empilha
\$<&<id>	/id\$	reduz
\$<&>	/id\$	reduz
\$<	/id\$	empilha
\$</<	id\$	empilha
\$</<id>	\$	reduz
\$</>	\$	reduz
\$E	\$	SUCESSO

18

ASA: precedência de operadores

- Exercício: reconheça a expressão (id)

$\langle E \rangle ::= \langle E \rangle / \langle T \rangle \mid \langle T \rangle$
 $\langle T \rangle ::= \langle T \rangle \& \langle F \rangle \mid \langle F \rangle$
 $\langle F \rangle ::= (\langle E \rangle) \mid \text{id}$

Pilha	Cadeia	Regra
\$	(id)\$	

Tabela sintática

	id	/	&	()	\$
id		>	>		>	>
/	<	>	<	<	>	>
&	<	>	>	<	>	>
(<	<	<	<	=	
)		>	>		>	>
\$	<	<	<	<		

19

ASA: precedência de operadores

- Exercício: reconheça a expressão (id)

$\langle E \rangle ::= \langle E \rangle / \langle T \rangle \mid \langle T \rangle$
 $\langle T \rangle ::= \langle T \rangle \& \langle F \rangle \mid \langle F \rangle$
 $\langle F \rangle ::= (\langle E \rangle) \mid \text{id}$

Pilha	Cadeia	Regra
\$<	(id)\$	empilha
\$<(<	id)\$	empilha
\$<(<id>)\$	reduz
\$<(<id>=<)\$	empilha
\$<(<id>=<id>=<	\$	reduz
\$E	\$	SUCESSO

Tabela sintática

	id	/	&	()	\$
id		>	>		>	>
/	<	>	<	<	>	>
&	<	>	>	<	>	>
(<	<	<	<	=	
)		>	>		>	>
\$	<	<	<	<		

20