

Compiladores Aula 09

Celso Olivete Júnior

olivete@fct.unesp.br



Análise sintática ascendente

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



- Simples e eficiente
- □ Aplicada, principalmente, para o <u>reconhecimento</u> <u>de expressões</u>
- ☐ Subclasse de gramáticas
 - > Gramáticas de (precedência de) operadores
 - 1. Não há símbolos não-terminais adjacentes
 - 2. Não há produções que derivam a cadeia nula



☐ Exemplo: a gramática abaixo não é de precedência de operadores – três não-terminais consecutivos do lado direito

```
<E> ::= <E><O><E> | (<E>) | id
<O> ::= + | -
```

☐ Transformando-a em gramática de operadores:



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

- 1. Relações de precedência → Considere os terminais a e b
 - 1.1. a < b significa que a tem precedência menor do que b
 - 1.2. a=b significa que a e b têm a mesma precedência
 - 1.3. a>b significa que a tem precedência maior do que b
- 2. Durante a análise ascendente, na pilha:
 - 2.1. < identifica o limite esquerdo do lado direito do handle
 - 2.2. = indica que os terminais envolvidos pertencem ao mesmo handle
 - 2.3. > identifica o limite direito do lado direito do handle



- 🗷 Tabela sintática → usando precedência de operadores
 - \Box Matriz quadrada que relaciona todos os terminais da gramática e o símbolo delimitador utilizado (\$ ou λ ou ε)
 - ☐ Primeira linha da tabela: terminais da cadeia sendo analisada
 - ☐ Primeira coluna da tabela: terminais do topo da pilha

	id	+	*	\$	→ cadeia
id		>	>	>	
+	<	>	<	>	
*	<	>	>	>	
\$	<	<	<	ok	

pilha



- 🗖 Regras para o 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
 - 2. 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

- 1. Se a < b ou a = b, então empilha b
- Obs: a→topo da pilha e b→terminal em análise
- 2. Se a>b, então 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

Exercício:

(id)

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

Pilha	Cadeia	Regra
\$	(id) \$	
\$<	(id) \$	Empilha
\$ <(id) \$	Reduz
\$ <(<	id) \$	Empilha
\$ <(<id< td=""><td>)\$</td><td>Reduz</td></id<>)\$	Reduz
\$ <(<id>></id>)\$	Reduz
\$ <()\$	Empilha
\$ <(=)	\$	Reduz
\$ <()>	\$	Reduz
\$E	\$	Aceito

- 1. Se a<b ou a=b, então se empilha b Obs: a→topo da pilha e b→terminal em análise
- 2. 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

Pilha

Cadeia

id&id/id\$

Regra

	_		~	1/
/	Eyemn	י אוי	expressões	lodicas
_		10.	CAPICSSUCS	logicas

id&id/id

<F> ::= (<E>) | id

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

→ cadeia

- 1. Se a<b ou a=b, então se empilha b Obs: a→topo da pilha e b→terminal em análise
- 2. 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

<e> ::= <e>/<t> <t></t></t></e></e>
<t> ::= <t>&<f> <f></f></f></t></t>
<f> ::= (<e>) id</e></f>

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

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

- 1. Se a<b ou a=b, então se empilha b Obs: a→topo da pilha e b→terminal em análise
- 2. 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

/	_	•	~	. / .
	Evemn	In av	nracchad	s lógicas
$\overline{}$		IU. CA	pi	ologicas

<e>::= <e>/<t> </t></e></e>	<t></t>
<t>::= <t>&<f> </f></t></t>	<f></f>
<f> ::= (<e>) id</e></f>	

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

Pilha	Cadeia	Regra
\$	id&id/id <mark>\$</mark>	
\$<	id&id/id <mark>\$</mark>	Empilha
\$ <id>></id>	&id/id\$	Reduz

- 1. Se a<b ou a=b, então se empilha b Obs: a→topo da pilha e b→terminal em análise
- 2. 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

<E> ::= <E>/<T> | <T> <T> ::= <T>&<F> | <F> <F> ::= (<E>) | id

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

Pilha	Cadeia	Regra
\$	id&id/id <mark>\$</mark>	
\$<	id&id/id <mark>\$</mark>	Empilha
\$ <id>></id>	&id/id\$	Reduz
\$<	&id/id\$	Empilha

- 1. Se a<b ou a=b, então se empilha b Obs: a→topo da pilha e b→terminal em análise
- 2. 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

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

Pilha	Cadeia	Regra
\$	id&id/id\$	
\$<	id&id/id\$	Empilha
\$ <id>></id>	&id/id\$	Reduz
\$< \$<&<	&id/id\$	Empilha
\$ <&<	id/id\$	Empilha

- 1. Se a<b ou a=b, então se empilha b Obs: a→topo da pilha e b→terminal em análise
- 2. 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

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

Pilha	Cadeia	Regra
\$	id&id/id\$	
\$<	id&id/id <mark>\$</mark>	Empilha
\$ <id>></id>	&id/id\$	Reduz
\$<	&id/id\$	Empilha
\$ <&<	id/id\$	Empilha
\$<& <id>></id>	/id\$	Reduz

- 1. Se a<b ou a=b, então se empilha b Obs: a→topo da pilha e b→terminal em análise
- 2. 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

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

Pilha	Cadeia	Regra
\$	id&id/id <mark>\$</mark>	
\$<	id&id/id <mark>\$</mark>	Empilha
\$ <id>></id>	&id/id\$	Reduz
\$<	&id/id\$	Empilha
\$ <&<	id/id\$	Empilha
\$<& <id>></id>	/id\$	Reduz
\$ <&>	/id\$	Reduz

- 1. Se a<b ou a=b, então se empilha b Obs: a→topo da pilha e b→terminal em análise
- 2. 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

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

Pilha	Cadeia	Regra
\$	id&id/id\$	
\$ <	id&id/id\$	Empilha
\$ <id>></id>	&id/id\$	Reduz
\$ <	&id/id\$	Empilha
\$ <&<	id/id\$	Empilha
\$<& <id>></id>	/id\$	Reduz
\$ <&>	/id\$	Reduz
\$<	/id\$	Empilha

- 1. Se a<b ou a=b, então se empilha b Obs: a→topo da pilha e b→terminal em análise
- 2. 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

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

Pilha	Cadeia	Regra
\$	id&id/id\$	
\$<	id&id/id\$	Empilha
\$ <id>></id>	&id/id\$	Reduz
\$<	&id/id\$	Empilha
\$ <&<	id/id\$	Empilha
\$ <& <id></id>	/id\$	Reduz
\$ <&>	/id\$	Reduz
\$< \$ <</td <td>/id\$</td> <td>Empilha</td>	/id\$	Empilha
\$ <</td <td>id\$</td> <td>Empilha</td>	id\$	Empilha

- 1. Se a<b ou a=b, então se empilha b Obs: a→topo da pilha e b→terminal em análise
- 2. 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

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

Pilha	Cadeia	Regra
\$	id&id/id\$	
\$<	id&id/id\$	Empilha
\$ <id>></id>	&id/id\$	Reduz
\$<	&id/id\$	Empilha
\$ <&<	id/id\$	Empilha
\$<& <id></id>	/id\$	Reduz
\$ <&>	/id\$	Reduz
\$<	/id\$	Empilha
\$ <</td <td>id\$</td> <td>Empilha</td>	id\$	Empilha
\$ <id	\$	Reduz

pilha

- 1. Se a<b ou a=b, então se empilha b Obs: a→topo da pilha e b→terminal em análise
- 2. 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

<E> ::= <E>/<T> | <T> <T> ::= <T>&<F> | <F> <F> ::= (<E>) | id

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

Pilha Cadeia Regra \$ id&id/id\$ \$< id&id/id\$ Empilha **\$**<i**d**> &id/id\$ Reduz \$< &id/id\$ Empilha \$<&< id/id\$ Empilha \$<&<id> /id\$ Reduz **\$**<&> /id\$ Reduz \$< /id\$ Empilha \$</< id\$ Empilha \$</<id> **\$** Reduz **\$**</> \$ Reduz

- 1. Se a<b ou a=b, então se empilha b Obs: a→topo da pilha e b→terminal em análise
- 2. 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

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

Pilha	Cadeia	Regra
\$	id&id/id\$	
\$<	id&id/id\$	Empilha
\$ <id>></id>	&id/id\$	Reduz
\$<	&id/id\$	Empilha
\$ <&<	id/id\$	Empilha
\$ <& <id>></id>	/id\$	Reduz
\$ <&>	/id\$	Reduz
\$<	/id\$	Empilha
\$ <</td <td>id\$</td> <td>Empilha</td>	id\$	Empilha
\$ <id	\$	Reduz
\$	\$	Reduz
\$E	\$	Sucesso

↓pilha



Algorimo do ASA de precedência de operadores

Seja S o símbolo inicial da gramática, a o símbolo terminal mais ao topo da pilha e b o primeiro símbolo da cadeia de entrada

repita

```
se ($S é o topo da pilha e $ é o primeiro símbolo da cadeia)
  então SUCESSO
senão se (a<b ou a=b)
  então empilha b
  senão se (a>b)
  então desempilha até haver < entre o terminal do topo e o último desempilhado
  senão ERRO</pre>
```



2 métodos para construção da tabela sintática

Intuitivo: baseado no conhecimento da precedência e associatividade dos operadores

Mecânico: obtêm a tabela diretamente da gramática



🗖 Método intuitivo

- □ Para 2 operadores quaisquer x e y
 - 1. Se x tem maior precedência do que y, então tem-se x (na pilha) > y (na cadeia) e y (na pilha) < x (na cadeia)
 - ✓ Exemplo: como o operador de * tem maior precedência que o operador de +, então *>+ e +<*
 - 2. Se x e y têm precedência igual (ou são iguais) e são associativos à esquerda, então tem-se x>y e y>x; e se são associativos à direita, então tem-se x<y e y<x
 - Exemplo: como * e / têm a mesma precedência e são associativos à esquerda, tem-se *>/ e />*; como o operador de exponenciação ** é associativo à direita, tem-se **<**</p>



- Método intuitivo
- ☐ Para 2 operadores quaisquer x e y
 - 3. As relações entre os operadores e os demais símbolos terminais (operandos e delimitadores) são fixas
 - Para qualquer operador x, tem-se x>\$, \$<x, x<id, id>x, x<(,(<x,x>)e)>x

4. As relações entre os operandos também são fixas \checkmark (<(,)>), id>), \$<(, (=),)>\$, id>\$, \$<id, (<id



Método intuitivo

Exemplo: construir a tabela sintática para a gramática abaixo

$$\langle E \rangle ::= \langle E \rangle + \langle E \rangle \mid \langle E \rangle^* \langle E \rangle \mid \langle E \rangle^* | (\langle E \rangle) \mid id$$

- ** tem maior precedência e é associativo à direita;
- > * tem precedência intermediária e é associativo à esquerda;
- > + tem menor precedência e é associativo à esquerda



Método intuitivo

☐ Exemplo: construir a tabela sintática para a gramática abaixo

- ** tem maior precedência e é associativo à direita;
- * tem precedência intermediária e é associativo à esquerda;
- > + tem menor precedência e é associativo à esquerda
- Se x tem maior precedência do que y, então tem-se x (na pilha) > y (na cadeia) e y (na pilha) < x (na cadeia)
- 2. Se x e y têm precedência igual (ou são iguais) e são associativos à esquerda, então tem-se x>y e y>x; se são associativos à direita, então tem-se x<y e y<x
- 3. As relações entre os operadores e os demais símbolos terminais (operandos e delimitadores) são fixas
- Para qualquer operador x, tem-se x>\$, \$<x, x<id, id>x, x<(, (<x, x>) e)>x
- 4. As relações entre os operandos também são fixas \checkmark (<(,)>), id>), \$<(, (=),)>\$, id>\$, \$<id, (<id

+	*	**	()	id	\$
	+	+ *	+ * **	+ * ** (+ * ** ()	+ * ** () id



Método intuitivo

☐ Exemplo: construir a tabela sintática para a gramática abaixo

- ** tem maior precedência e é associativo à direita;
- * tem precedência intermediária e é associativo à esquerda;
- > + tem menor precedência e é associativo à esquerda
- Se x tem maior precedência do que y, então tem-se x (na pilha) > y (na cadeia) e y (na pilha) < x (na cadeia)
- 2. Se x e y têm precedência igual (ou são iguais) e são associativos à esquerda, então tem-se x>y e y>x; se são associativos à direita, então tem-se x<y e y<x
- 3. As relações entre os operadores e os demais símbolos terminais (operandos e delimitadores) são fixas
- Para qualquer operador x, tem-se x>\$, \$<x, x<id, id>x, x<(, (<x, x>) e)>x
- 4. As relações entre os operandos também são fixas \checkmark (<(,)>), id>), \$<(, (=),)>\$, id>\$, \$<id, (<id

		+	*	**	()	id	\$
	+	>	<	<	<	>	<	>
1	*							
	**							
	(
)							
	id							
	\$							
ı								



Método intuitivo

☐ Exemplo: construir a tabela sintática para a gramática abaixo

- ** tem maior precedência e é associativo à direita;
- * tem precedência intermediária e é associativo à esquerda;
- > + tem menor precedência e é associativo à esquerda
- Se x tem maior precedência do que y, então tem-se x (na pilha) > y (na cadeia) e y (na pilha) < x (na cadeia)
- 2. Se x e y têm precedência igual (ou são iguais) e são associativos à esquerda, então tem-se x>y e y>x; se são associativos à direita, então tem-se x<y e y<x
- 3. As relações entre os operadores e os demais símbolos terminais (operandos e delimitadores) são fixas
- \checkmark Para qualquer operador x, tem-se x>\$, \$<x, x<id, id>x, x<(, (<x, x>) e)>x
- 4. As relações entre os operandos também são fixas \checkmark (<(,)>), id>), \$<(, (=),)>\$, id>\$, \$<id, (<id

	+	*	**	()	id	\$
+	>	<	<	<	>	<	>
*	>	>	<	<	>	<	>
**							
(
)							
id							
\$							



Método intuitivo

☐ Exemplo: construir a tabela sintática para a gramática abaixo

- ** tem maior precedência e é associativo à direita;
- * tem precedência intermediária e é associativo à esquerda;
- > + tem menor precedência e é associativo à esquerda
- Se x tem maior precedência do que y, então tem-se x (na pilha) > y (na cadeia) e y (na pilha) < x (na cadeia)
- 2. Se x e y têm precedência igual (ou são iguais) e são associativos à esquerda, então tem-se x>y e y>x; se são associativos à direita, então tem-se x<y e y<x
- 3. As relações entre os operadores e os demais símbolos terminais (operandos e delimitadores) são fixas
- Para qualquer operador x, tem-se x>\$, \$<x, x<id, id>x, x<(, (<x, x>) e)>x
- 4. As relações entre os operandos também são fixas \checkmark (<(,)>), id>), \$<(, (=),)>\$, id>\$, \$<id, (<id

		+	*	**	()	id	\$
	+	>	<	<	<	>	<	>
1	*	>	>	<	<	>	<	>
	**	>	>	<	<	>	<	>
	(
)							
	id							
	\$							



Método intuitivo

☐ Exemplo: construir a tabela sintática para a gramática abaixo

- ** tem maior precedência e é associativo à direita;
- * tem precedência intermediária e é associativo à esquerda;
- > + tem menor precedência e é associativo à esquerda
- Se x tem maior precedência do que y, então tem-se x (na pilha) > y (na cadeia) e y (na pilha) < x (na cadeia)
- 2. Se x e y têm precedência igual (ou são iguais) e são associativos à esquerda, então tem-se x>y e y>x; se são associativos à direita, então tem-se x<y e y<x
- 3. As relações entre os operadores e os demais símbolos terminais (operandos e delimitadores) são fixas
- Para qualquer operador x, tem-se x>\$, \$<x, x<id, id>x, x<(, (<x, x>) e)>x
- 4. As relações entre os operandos também são fixas \checkmark (<(,)>), id>), \$<(, (=),)>\$, id>\$, \$<id, (<id

	+	*	**	()	id	\$
+	>	<	<	<	>	<	>
*	>	>	<	<	>	<	>
**	>	>	<	<	>	<	>
(<	<	<	<	=	<	
)							
id							
\$							



Método intuitivo

☐ Exemplo: construir a tabela sintática para a gramática abaixo

- ** tem maior precedência e é associativo à direita;
- * tem precedência intermediária e é associativo à esquerda;
- > + tem menor precedência e é associativo à esquerda
- Se x tem maior precedência do que y, então tem-se x (na pilha) > y (na cadeia) e y (na pilha) < x (na cadeia)
- 2. Se x e y têm precedência igual (ou são iguais) e são associativos à esquerda, então tem-se x>y e y>x; se são associativos à direita, então tem-se x<y e y<x
- 3. As relações entre os operadores e os demais símbolos terminais (operandos e delimitadores) são fixas
- \checkmark Para qualquer operador x, tem-se x>\$, \$<x, x<id, id>x, x<(, (<x, x>) e)>x
- 4. As relações entre os operandos também são fixas \checkmark (<(,)>), id>), \$<(, (=),)>\$, id>\$, \$<id, (<id

	+	*	**	()	id	\$
+	>	<	<	<	>	<	>
*	>	>	<	<	>	<	>
**	>	>	<	<	>	<	>
(<	<	<	<	=	<	
)	>	>	>		>		>
id							
\$							



Método intuitivo

☐ Exemplo: construir a tabela sintática para a gramática abaixo

- ** tem maior precedência e é associativo à direita;
- * tem precedência intermediária e é associativo à esquerda;
- > + tem menor precedência e é associativo à esquerda
- Se x tem maior precedência do que y, então tem-se x (na pilha) > y (na cadeia) e y (na pilha) < x (na cadeia)
- Se x e y têm precedência igual (ou são iguais) e são associativos à esquerda, então tem-se x>y e y>x; se são associativos à direita, então tem-se x<y e y<x
- 3. As relações entre os operadores e os demais símbolos terminais (operandos e delimitadores) são fixas
- Para qualquer operador x, tem-se x>\$, \$<x, x<id, id>x, x<(, (<x, x>) e)>x
- 4. As relações entre os operandos também são fixas \checkmark (<(,)>), id>), \$<(, (=),)>\$, id>\$, \$<id, (<id

	+	*	**	()	id	\$
+	>	<	<	<	>	<	>
*	>	>	<	<	>	<	>
**	>	>	<	<	>	<	>
(<	<	<	<	=	<	
)	>	>	>		>		>
id	>	>	>		>		>
\$							
"							



Método intuitivo

☐ Exemplo: construir a tabela sintática para a gramática abaixo

- ** tem maior precedência e é associativo à direita;
- * tem precedência intermediária e é associativo à esquerda;
- > + tem menor precedência e é associativo à esquerda
- Se x tem maior precedência do que y, então tem-se x (na pilha) > y (na cadeia) e y (na pilha) < x (na cadeia)
- 2. Se x e y têm precedência igual (ou são iguais) e são associativos à esquerda, então tem-se x>y e y>x; se são associativos à direita, então tem-se x<y e y<x
- 3. As relações entre os operadores e os demais símbolos terminais (operandos e delimitadores) são fixas
- \checkmark Para qualquer operador x, tem-se x>\$, \$<x, x<id, id>x, x<(, (<x, x>) e)>x
- 4. As relações entre os operandos também são fixas \checkmark (<(,)>), id>), \$<(, (=),)>\$, id>\$, \$<id, (<id

	+	*	**	()	id	\$
+	>	<	<	<	>	<	>
*	>	>	<	<	>	<	>
**	>	>	<	<	>	<	>
(<	<	<	<	=	<	
)	>	>	>		>		>
id	>	>	>		>		>
\$	<	<	<	<		<	ok



- 🗖 Método mecânico: aplicável para <u>gramáticas não ambíguas</u>
- ☐ Para os terminais a e b
 - 1. a=b se $\alpha a\beta b\gamma$ é lado direito de produção e β é ϵ ou um único símbolo não terminal
 - 2. a
b se α aX β é lado direito de produção e X produz γ b δ e γ é ϵ ou
 um único símbolo não terminal
 - 3. \$ se S produz γ b δ e γ é ϵ ou um único símbolo não terminal
 - 4. a>b se $\alpha Xb\beta$ é lado direito de produção e X produz $\gamma a\delta$ e δ é ϵ ou um único símbolo não terminal
 - 5. a>\$ se S produz $\gamma a \delta$ e δ é ϵ ou um único símbolo não terminal



Método mecânico → em outras palavras...

Regra 1 – aX Um terminal a seguido imediatamente de um não terminal x tem precedência menor do que os primeiros símbolos terminais deriváveis a partir de x (precedidos de ϵ ou um não terminal)

Regra 2 – **Xa** Todos os últimos terminais que podem ser derivados a partir de um não terminal **X** (seguidos de ε ou um não terminal) têm **precedência maior** do que um terminal que segue imediatamente a **X**

$$Xa \rightarrow Ultimos(X) > a$$

Regra 3 – aXb Para computar =, procurar aXb nos lados direitos das produções, onde a ou b é ε ou um terminal, e fazer a=b

Regra 4 – Delimitadores \$ tem **precedência menor** do que todos os primeiros terminais deriváveis a partir do símbolo inicial da gramática. Todos os últimos terminais derivados a partir do símbolo inicial da gramática têm **precedência maior** do que \$



Exemplo: construir a tabela sintática para a gramática abaixo

☐ Inicialmente, se a gramática for ambígua, deve-se **eliminar a ambiguidade** da gramática (mantendo a precedência e a associatividade dos operadores)

```
<E>::= <E>+<T> | <T>
<T> ::= <T>*<F> | <F>
<F> ::= <P>***<F> | <P>
<P> ::= (<E>) | id
```



☐ Determinam-se, para cada não terminal, os primeiros e últimos terminais possíveis de ocorrerem em uma cadeia derivada a partir do não terminal

	Primeiros	Últimos
E	+ * ** (id	+ * **) id
Т	* ** (id	* **) id
F	** (id	**) id
Р	(id) id



- Para computar <, procurar pares aX nos lados direitos de produção;</p>
 - > tem-se que a tem menor precedência do que qualquer primeiro

terminal derivado a partir de X

> Pares:

> Relações:

/ **< {**,(,id}</pre>

√ (< {+,*,**,(,id}

ilha cadeia

	Primeiros	Últimos
E	+ * ** (id	+ * **) id
T	* ** (id	* **) id
F	** (id	**) id
P	(id) id

	+	*	**	()	id	\$
+		<	<	<		<	
*			<	<		<	
**			<	<		<	
(<	<	<	<		<	
)							
id							
\$							



- Para computar >, procurar pares Xb nos lados direitos de produção;
 - > tem-se que qualquer último terminal derivado de X tem precedência

maior do que b

> Pares:

> Relações:

	Primeiros	Últimos
E	+ * ** (id	+ * **) id
Т	* ** (id	* **) id
F	** (id	**) id
Р	(id) id

<e>::= <e>+<t> <t></t></t></e></e>
<t>::=<t>*<f> <f></f></f></t></t>
<f> ::= <p>**<f> <p></p></f></p></f>
<p> ::= (<e>) id</e></p>

	+	*	**	()	id	\$
+	>	<	<	<	>	<	
*	>	>	<	<	>	<	
**	>	>	<	<	>	<	
(<	<	<	<		<	
)	>	>	>		>		
id	>	>	>		>		
ç							

cadeia



 \square Para computar =, procurar $a\beta b$ nos lados direitos das produções,

onde β é ϵ ou um terminal, e fazer a=b

➤ Dado o lado direito (E), tem-se (=) → aXb

Delimitadores

\$ tem precedência menor do que todos os primeiros terminais deriváveis a partir do símbolo inicial da gramática

		+	*	**	()	id	\$
)	+	>	<	<	<	>	<	>
	*	>	>	<	<	>	<	>
	**	>	>	<	<	>	<	>
	(<	<	<	<	=	<	
)	>	>	>		>		>
	id	>	>	>		>		>
	\$	<	<	<	<		<	

+ * ** (id

* ** (id

** (id

(id

+ * * *) id

* **) id

**) id

) id

Todos os últimos terminais derivados a partir do símbolo inicial da gramática têm precedência maior do que \$

oilha

cadeia



- Relembrando como usar a 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

□Seja a o terminal <u>mais ao topo</u> 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

	+	*	**	()	id	\$	-
+	>	<	<	<	>	<	>	
*	>	>	<	<	>	<	>	
**	>	>	<	<	>	<	>	
(<	<	<	<	=	<		
)	>	>	>		>		>	
id	>	>	>		>		>	
\$	<	<	<	<		<		

🔾 cadeia

<e> ::= <e>+<t> <t></t></t></e></e>
<t>::= <t>*<f> <f></f></f></t></t>
<f> ::= <p>**<f> <p></p></f></p></f>
<p> ::= (<e>) id</e></p>
Compiladore

Pilha	Cadeia	Regra
\$<	(id*id)\$	Empilha
\$<(id*id)\$	Empilha
\$<(<id>></id>	*id)\$	Reduz
\$<(*	id) \$	Empilha
\$<(* <id>></id>)\$	Empilha
\$<(*)\$	Reduz
\$<()\$	Empilha
\$<()	\$	Reduz
\$<()>	\$	Reduz
\$E	\$	Aceita



Exercício:

1. Construir a tabela sintática para a gramática abaixo utilizando o método mecânico e o intuitivo. Reconheça a cadeia id/(id/id)

```
<E> ::= <E>/<T> | <T>
<T> ::= <T>&<F> | <F>
<F> ::= (<E>) | id
```

2. Construir a tabela sintática (método mecânico) para a gramática abaixo pelo método mecânico e o intuitivo

2.1. Utilizando a tabela construída anteriormente, reconheça a cadeia (a*b)



→ 2. Resposta:

$$S \rightarrow (SOS)|a|b$$

 $O \rightarrow + |*$

Transformando a gramática → não pode ocorrer três não terminais consecutivos em uma gramática de operadores

```
<E> ::= <E>+<T> | <T>
<T> ::= <T>*<F> | <F>
<F> ::= (<E>) | a | b
```

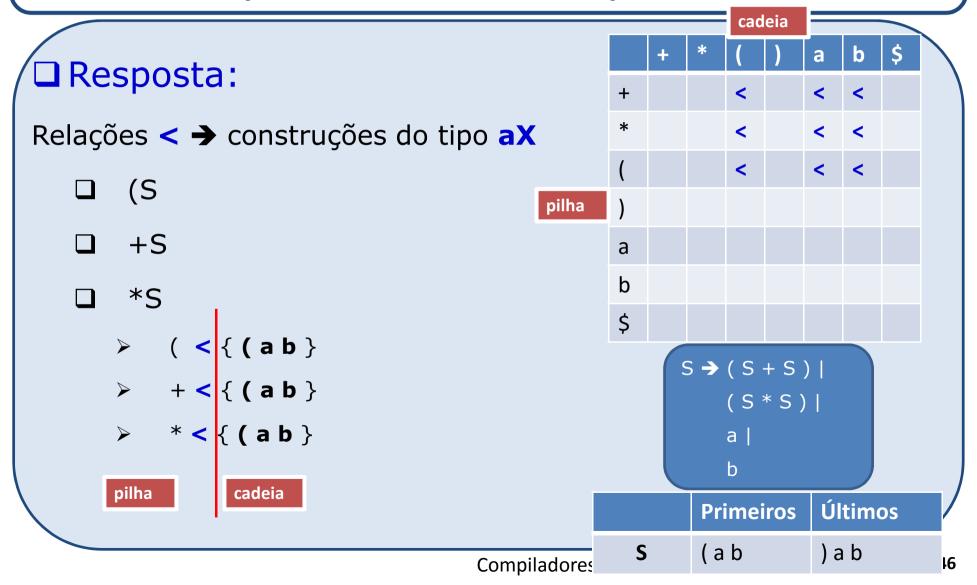


☐ Resposta:

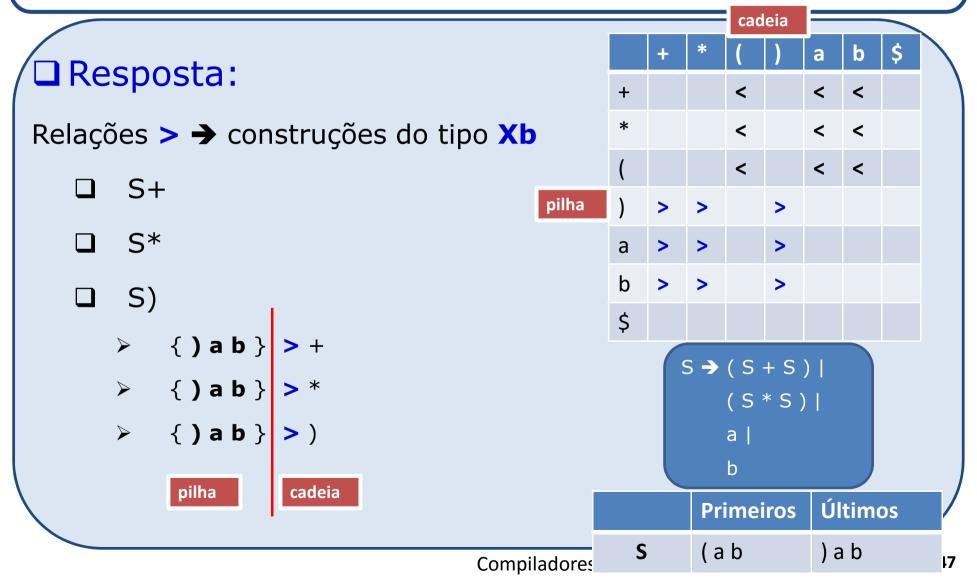
Obtendo os conjuntos primeiros e últimos

	Primeiros	Últimos
S	(a b) a b

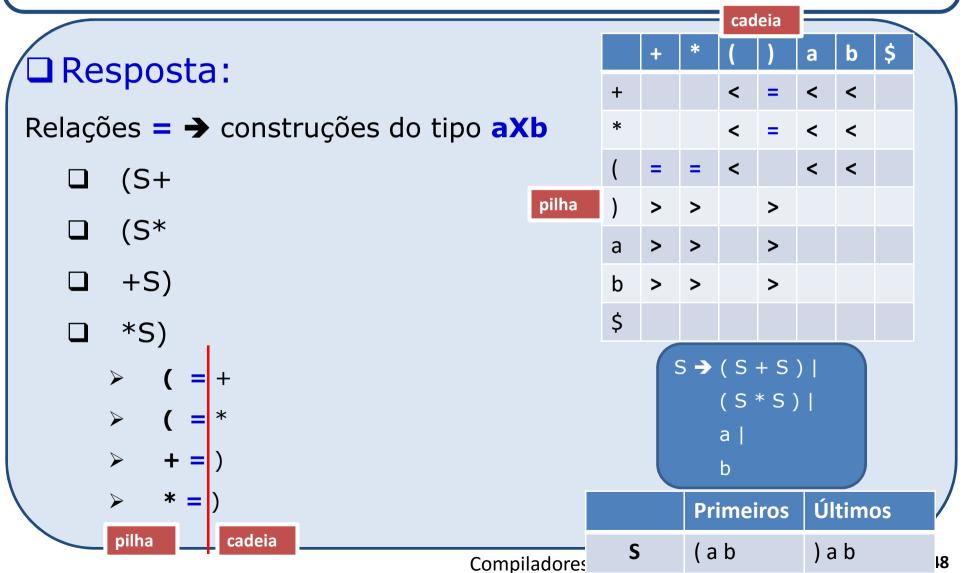














→ Resposta:

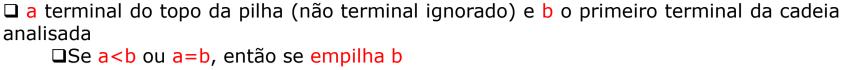
Delimitadores

- > \$ < primeiros de S
- √ Últimos de S > \$

✓ {) a b } > \$

	<u> </u>								
	,			cac	leia				
		+	*	()	а	b	\$	
	+			<	=	<	<		
	*			<	=	<	<		
	(=	=	<		<	<		
pilha)	>	>		>			>	
	а	>	>		>			>	
	b	>	>		>			>	
	\$			<		<	<		
			S →		+ S * S)				
			Pr	ime	iros	Ú	tim	os	

Primeiros Últimos
S (ab) a b



une

□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

☐ 2. Resposta:

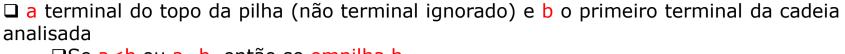
Reconhecendo a cadeia

(a*b)

	+	*	()	а	b	\$
+			<	=	<	<	
*			<	=	<	<	
(=	=	<		<	<	
)	>	>		>			>
а	>	>		>			>
b	>	>		>			>
\$			<		<	<	

Pilha	Cadeia	Regra
\$	(a*b) \$	

 $S \rightarrow (S + S) | (S * S) | a | b$



- □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

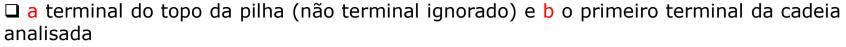
Reconhecendo a cadeia

(a*b)

	+	*	()	а	b	\$
+			<	=	<	<	
*			<	=	<	<	
(=	=	<		<	<	
)	>	>		>			>
а	>	>		>			>
b	>	>		>			>
\$			<		<	<	

Pilha	Cadeia	Regra
\$<	(a*b)\$	Empilha

 $S \rightarrow (S + S) | (S * S) | a | b$





□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

☐ 2. Resposta:

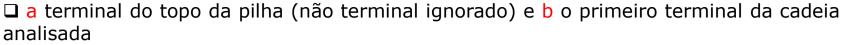
Reconhecendo a cadeia

(a*b)

	+	*	()	а	b	\$
+			<	=	<	<	
*			<	=	<	<	
(=	=	<		<	<	
)	>	>		>			>
а	>	>		>			>
b	>	>		>			>
\$			<		<	<	

Pilha	Cadeia	Regra
\$< \$<(<	(a*b) \$	Empilha
\$ <(<	a*b) \$	Empilha

 $S \rightarrow (S + S) | (S * S) | a | b$





□Se a>b, então se procura o lado direito do handle na pilha e o substitui pelo seu lado esquerdo

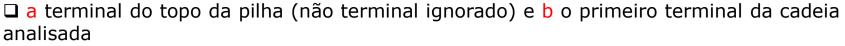
□O lado direito do *handle* estará delimitado na pilha pelos símbolos < e > □Os não terminais não precisam aparecer, mas se deve saber que foram produzidos e que seus derivados correspondentes foram consumidos

☐ 2. Resposta:

Reconhecendo a cadeia

	+	*	()	а	b	\$
+			<	=	<	<	
*			<	=	<	<	
(=	=	<		<	<	
)	>	>		>			>
а	>	>		>			>
b	>	>		>			>
\$			<		<	<	

Pilha	Cadeia	Regra
\$<	(a*b) <mark>\$</mark>	Empilha
\$< \$<(< \$<(<a>>	a*b) <mark>\$</mark>	Empilha
\$ <(<a>>	*b) \$	Reduz





- □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

Reconhecendo a cadeia

	+	*	()	а	b	\$
+			<	=	<	<	
*			<	=	<	<	
(=	=	<		<	<	
)	>	>		>			>
а	>	>		>			>
b	>	>		>			>
\$			<		<	<	

Pilha	Cadeia	Regra
\$<	(a*b) \$	Empilha
\$< \$<(< \$<(<a>> \$<(=	a*b) \$	Empilha
\$ <(<a>>	*b) \$	Reduz
\$ <(=	*b) \$	Empilha

- □ a terminal do topo da pilha (não terminal ignorado) e b o primeiro terminal da cadeia 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

Reconhecendo a cadeia

	+	*	()	а	b	\$
+			<	=	<	<	
*			<	=	<	<	
(=	=	<		<	<	
)	>	>		>			>
а	>	>		>			>
b	>	>		>			>
\$			<		<	<	

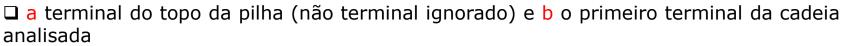
Pilha	Cadeia	Regra
\$<	(a*b) \$	Empilha
\$< \$<(< \$<(<a>> \$<(= \$<(=*<	a*b) \$	Empilha
\$<(<a>>	*b) \$	Reduz
\$<(=	*b) \$	Empilha
\$ <(=*<	b) \$	Empilha

- □ a terminal do topo da pilha (não terminal ignorado) e b o primeiro terminal da cadeia 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

Reconhecendo a cadeia

	+	*	()	а	b	\$
+			<	=	<	<	
*			<	=	<	<	
(=	=	<		<	<	
)	>	>		>			>
а	>	>		>			>
b	>	>		>			>
\$			<		<	<	

Pilha	Cadeia	Regra
\$<	(a*b) \$	Empilha
\$<(<	a*b) \$	Empilha
\$<(<a>>	*b) \$	Reduz
\$<(=	*b) \$	Empilha
\$<(=*<	b) \$	Empilha
\$ <(=*)\$	Reduz



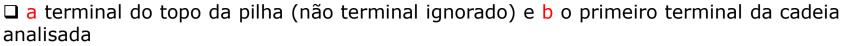


- □Se a>b, então se procura o lado direito do handle na pilha e o substitui pelo seu lado esquerdo
 - □O lado direito do *handle* estará delimitado na pilha pelos símbolos < e > □Os não terminais não precisam aparecer, mas se deve saber que foram produzidos e que seus derivados correspondentes foram consumidos

Reconhecendo a cadeia

	+	*	()	а	b	\$
+			<	=	<	<	
*			<	=	<	<	
(=	=	<		<	<	
)	>	>		>			>
а	>	>		>			>
b	>	>		>			>
\$			<		<	<	

Pilha	Cadeia	Regra
\$<	(a*b) \$	Empilha
\$<(<	a*b) \$	Empilha
\$<(<a>>	*b) \$	Reduz
\$<(=	*b) \$	Empilha
\$<(=*<	b) \$	Empilha
\$<(=* \$<(=*=)\$	Reduz
\$ <(=*=)\$	Empilha



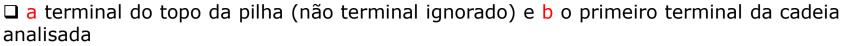


- □Se a>b, então se procura o lado direito do handle na pilha e o substitui pelo seu lado esquerdo
 - □O lado direito do *handle* estará delimitado na pilha pelos símbolos < e > □Os não terminais não precisam aparecer, mas se deve saber que foram produzidos e que seus derivados correspondentes foram consumidos

Reconhecendo a cadeia

	+	*	()	а	b	\$
+			<	=	<	<	
*			<	=	<	<	
(=	=	<		<	<	
)	>	>		>			>
а	>	>		>			>
b	>	>		>			>
\$			<		<	<	

Pilha	Cadeia	Regra
\$ <	(a*b) \$	Empilha
\$ <(<	a*b) \$	Empilha
\$<(<a>>	*b)\$	Reduz
\$ <(=	*b) \$	Empilha
\$ <(=*<	b) \$	Empilha
\$<(=*)\$	Reduz
\$<(=*=)\$	Empilha
\$ <(=*=)	\$	Reduz





- □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

Reconhecendo a cadeia

	+	*	()	а	b	\$
+			<	=	<	<	
*			<	=	<	<	
(=	=	<		<	<	
)	>	>		>			>
а	>	>		>			>
b	>	>		>			>
\$			<		<	<	

Pilha	Cadeia	Regra
\$<	(a*b) \$	Empilha
\$ <(<	a*b) \$	Empilha
\$ <(<a>	*b) \$	Reduz
\$ <(=	*b) \$	Empilha
\$ <(=*<	b) \$	Empilha
\$ <(=*)\$	Reduz
\$ <(=*=)\$	Empilha
\$ <(=*=)>	\$	Reduz
\$ S	\$	Aceita



ASA - precedência de operadores <u>Exercício</u>

□ Reconheça a cadeia → a(a+b)

	+	*	()	а	b	\$
+			<	=	<	<	
*			<	=	<	<	
(=	=	<		<	<	
)	>	>		>			>
а	>	>		>			>
b	>	>		>			>
\$			<		<	<	

Pilha	Cadeia	Regra
\$	a(a+b) \$	