

# Tipos de Analizadores Sintácticos LR(k)

- LR(0)
- SLR(1): Simple LR(1)
- LALR(1): *Lookahead* LR(1)
- LR(1)

## Poder expresivo

$$\begin{array}{ccccccc} LR(0) & \subset & SLR(1) & \subset & LALR(1) & \subset & LR(1) \\ \uparrow & & & & & & \uparrow \\ \text{menor} & & & & & & \text{mayor} \end{array}$$

Los parsers LR se representan como las tablas *ir\_a* y *acción*:

- **Tabla acción** (Action): Mapea cada estado (del tope de la pila) y símbolo de entrada a una acción:
  - *desplazar i* (shift): desplazar y apilar el estado  $i$ .
  - *reducir*  $A \rightarrow \beta$ : reducir por la producción indicada.
  - *aceptar*: aceptar la cadena de entrada.
  - *error*: rechazar la cadena de entrada.
- **Tabla ir\_a** (GOTO): Mapea un estado y un símbolo no terminal a un nuevo estado.

# Conceptos para la construcción del AFD

## Item LR(0)

Producción con un pivote ( $\bullet$ ) en alguna posición del lado derecho.

Representa hasta dónde se vio una producción en el proceso de análisis sintáctico, y cómo se espera que continúe la cadena de entrada.

Por ejemplo, para la producción  $A \rightarrow BC$ , existen estos ítems LR(0):

$A \rightarrow \bullet BC$

$A \rightarrow B \bullet C$

$A \rightarrow BC \bullet$

Para la producción  $A \rightarrow \lambda$  el único ítem es:  $A \rightarrow \bullet$

# Conceptos para la construcción del AFD

## Clausura de un Conjunto de Items

```
 $J \leftarrow I$   
repeat  
  for ítem  $A \rightarrow \alpha \cdot \mathbf{B} \beta \in J$ , y producción  $\mathbf{B} \rightarrow \gamma$  de  $G$  do  
    agregar  $B \rightarrow \cdot \gamma$  a  $J$   
  end for  
until no se puedan agregar ítems a  $J$   
return  $J$ 
```

Por ejemplo, si tenemos

$$A \rightarrow BaC$$
$$B \rightarrow b$$
$$C \rightarrow c$$

La clausura de  $A \rightarrow \bullet BaC$  es

$$A \rightarrow \bullet BaC$$
$$B \rightarrow \bullet b$$

# Conceptos para la construcción del AFD

Función  $lr\_a(I$  conjunto de ítems,  $X$  símbolo)

```
 $J \rightarrow \emptyset$   
for cada ítem  $A \rightarrow \alpha \bullet X\beta$  en  $I$  do  
    agregar  $Clausura(\{A \rightarrow \alpha X \bullet \beta\})$  a  $J$   
end for  
return  $J$ 
```

Ítems de una gramática  $G$

```
 $C \leftarrow \{Clausura(\{S' \rightarrow \bullet S\})\}$   
repeat  
    for cada conjunto de ítems  $I$  en  $C$  y cada símbolo  $X$  do  
        agregar  $lr\_a(I, X)$  a  $C$   
    end for  
until no se puedan agregar más conjuntos de ítems a  $C$ 
```

## Construcción de las tablas

### 3 Construir la tabla acción:

```
if  $A \rightarrow \alpha \cdot a\beta$  está en  $I_i$ , y  $\delta(I_i, a) = I_j$ ,  $a \in V_T$  then  
    asignar desplazar  $j$  a  $accion[i, a]$   
end if  
if  $A \rightarrow \alpha \cdot$  está en  $I_i$ ,  $A \neq S'$  then  
    asignar reducir  $A \rightarrow \alpha$  a  $accion[i, a]$   
    LR(0): para todos los terminales  $a$   
    SLR(1): para  $a$  en  $Siguientes(A)$   
end if  
if  $S' \rightarrow S \cdot$  está en  $I_i$  then  
    asignar aceptar a  $accion[i, \$]$   
end if
```

## Construcción del Autómata

4 Construir la tabla `ir_a`:

Para cada no terminal  $A$ ,  $ir\_a[i, A] = j \Leftrightarrow \delta(I_i, A) = I_j$

5 Las entradas vacías de la tabla acción son consideradas *error*

6 El estado inicial es el que contiene el ítem  $S' \rightarrow \cdot S$

## Conflictos

Si la tabla acción tiene más de una entrada en algún casillero, entonces la gramática no es LR(0) / SLR(1).

Posibles conflictos:

- shift-reduce
- reduce-reduce



# Parsers LR(K) - Algoritmo

apilar  $s_0$

**loop**

$s \leftarrow$  tope de la pila

$a \leftarrow$  próximo símbolo apuntado en  $w\$$

**if**  $accion[s, a] = \text{desplazar } s' \text{ then}$

apilar  $s'$

avanzar al próximo símbolo de entrada

**else if**  $accion[s, a] = \text{reducir } A \rightarrow \beta \text{ then}$

sacar  $|\beta|$  símbolos de la pila

$s' \leftarrow$  tope de la pila

apilar  $ir\_a[s', A]$

**else if**  $accion[s, a] = \text{aceptar then}$

**return**

**else**

$error()$

**end if**

**end loop**

# Ejercicio 1

Sea  $G_1$  la siguiente gramática:

$$\begin{aligned} S &\rightarrow SA|A \\ A &\rightarrow (S)|() \end{aligned}$$

¿Es LR(0)? ¿Es SLR?

Analizar la siguiente cadena:  $(( ))()$

# Tabla acción LR(0) para G1

estado	(	)	\$
0	desplazar 3		
1	desplazar 3		aceptar
2	reducir $S \rightarrow A$	reducir $S \rightarrow A$	reducir $S \rightarrow A$
3	desplazar 3	desplazar 6	
4	reducir $S \rightarrow SA$	reducir $S \rightarrow SA$	reducir $S \rightarrow SA$
5	desplazar 3	desplazar 7	
6	reducir $A \rightarrow ()$	reducir $A \rightarrow ()$	reducir $A \rightarrow ()$
7	reducir $A \rightarrow (S)$	reducir $A \rightarrow (S)$	reducir $A \rightarrow (S)$

# Tabla ir\_a LR(0) para G1

estado	S	A
0	1	2
1		4
2		
3	5	2
4		
5		4
6		
7		

# Análisis de una cadena del lenguaje generado por G1

pila	entrada	acción
0	((()))\$	desplazar 3
03	()))\$	desplazar 3
033	)))\$	desplazar 6
0336	))\$	reducir $A \rightarrow ()$
032	)\$	reducir $S \rightarrow A$
035	)\$	desplazar 7
0357	()\$	reducir $A \rightarrow (S)$
02	()\$	reducir $S \rightarrow A$
01	()\$	desplazar 3
013	)\$	desplazar 6
0136	\$	reducir $A \rightarrow ()$
014	\$	reducir $S \rightarrow SA$
01	\$	aceptar

Derivación:

$S \Rightarrow SA \Rightarrow S() \Rightarrow A() \Rightarrow (S)() \Rightarrow (A)() \Rightarrow (())()$

## Ejercicio 2

Sea  $G_2$  la siguiente gramática:

$$E \rightarrow id \mid id(E) \mid E + id$$

¿Es LR(0)? ¿Es SLR?

Analizar la siguiente cadena:  $id(id + id)$

# Tabla acción SLR para G2

estado	id	(	)	+	\$
0	desp2				
1				desp3	aceptar
2		desp4	$r(E \rightarrow id)$	$r(E \rightarrow id)$	$r(E \rightarrow id)$
3	desp5				
4	desp2				
5			$r(E \rightarrow E + id)$	$r(E \rightarrow E + id)$	$r(E \rightarrow E + id)$
6			desp7	desp3	
7			$r(E \rightarrow id(E))$	$r(E \rightarrow id(E))$	$r(E \rightarrow id(E))$

Siguientes(E) = {), +, \$}

# Tabla ir\_a SLR para G2

estado	E
0	1
1	
2	
3	
4	6
5	
6	
7	



# Análisis de una cadena del lenguaje generado por G2

pila	entrada	acción
0	id(id+id)\$	desplazar 2
02	(id+id)\$	desplazar 4
024	id+id)\$	desplazar 2
0242	+id)\$	reducir $E \rightarrow id$
0246	+id)\$	desplazar 3
02463	id)\$	desplazar 5
024635	)\$	reducir $E \rightarrow E + id$
0246	)\$	desplazar 7
02467	\$	reducir $E \rightarrow id(E)$
01	\$	aceptar

Derivación:

$$S \Rightarrow id(E) \Rightarrow id(E + id) \Rightarrow id(id + id)$$