

Analizadores Sintácticos LR

OTRO tipo de analizador
ascendente

Análisis Sintáctico LR

- Técnica eficiente de análisis sintáctico ascendente.
- LR(k):
 - L: Examen de la Entrada de Izquierda a Derecha
 - R: Construye una derivación más derecha.
 - k: Número de símbolos de la entrada de examen por anticipado para tomar decisiones. (k=1, normalmente).

Análisis Sintáctico LR

- Comparación con LL(1): ASD y LR(1): ASA.

LL(k)	LR(k)
L: Examen de la entrada de Izquierda a derecha.	R: Construye una derivación más derecha.
L: Construye una derivación más izquierda.	R: Construye una derivación más derecha.
k: Número de símbolos de la entrada de examen por anticipado para tomar decisiones. (k=1, normalmente).	

Análisis Sintáctico LR

Tres Técnicas para construcción de tabla de análisis sintáctico LR:

- **SLR:** Single LR o LR Sencillo. El más fácil de implementar, pero el menos poderoso de los tres métodos.
- **LR Canónico:** Es el más poderoso y costoso de todos.
- **LALR:** Look Ahead LR o LR con examen por anticipado. Está en un punto intermedio.

Análisis Sintáctico por Despl/Red

Pila	Entrada	Acción
\$	w\$	
\$S	\$	Aceptar

w: cadena a reconocer

S: Símbolo no terminal de inicio de G

desplazar: transferir un terminal del frente de la entrada hacia el tope de la pila.

reducir: reemplazar una cadena α en la parte superior de la pila por un no terminal, dado que existe un mango $A \rightarrow \alpha$.

Generalidades

- Para realizar el Análisis Sintáctico LR se debe **aumentar la grámática.**
- Si S es el símbolo de inicio de la GIC, luego se agrega un nuevo símbolo inicial S' con la producción:

$$S' \rightarrow S$$

Por qué?

Ejemplo 1

Dada la **GIC**: $S \rightarrow (S)S \mid \in$

La GIC aumentada es:

$S' \rightarrow S$

$S \rightarrow (S)S \mid \in$

$w = ()$

No.	Pila	Entrada	Acción
1	\$	()\$	desplazar

()

Ejemplo 1

Dada la **GIC**: $S \rightarrow (S)S \mid \in$

La GIC aumentada es:

$S' \rightarrow S$

$S \rightarrow (S)S \mid \in$

$w = ()$

No.	Pila	Entrada	Acción
1	\$	()\$	desplazar
2	\$(ϵ))\$	reducir $S \rightarrow \in$

$(S) \Rightarrow ()$

Ejemplo 1

Dada la **GIC**: $S \rightarrow (S)S \mid \in$

La GIC aumentada es:

$S' \rightarrow S$

$S \rightarrow (S)S \mid \in$

$w = ()$

No.	Pila	Entrada	Acción
1	\$)\$	desplazar
2	\$()\$	reducir $S \rightarrow \in$
3	\$(S)\$	desplazar

$(S) \Rightarrow ()$

Ejemplo 1

Dada la **GIC**: $S \rightarrow (S)S \mid \in$

La GIC aumentada es:

$S' \rightarrow S$

$S \rightarrow (S)S \mid \in$

$w = ()$

No.	Pila	Entrada	Acción
1	\$	()\$	desplazar
2	\$()\$	reducir $S \rightarrow \in$
3	\$(S)\$	desplazar
4	\$(S)	\$	reducir $S \rightarrow \in$

$(S)S \Rightarrow (S) \Rightarrow ()$

Ejemplo 1

Dada la **GIC**: $S \rightarrow (S)S \mid \in$

La GIC aumentada es:

$S' \rightarrow S$

$S \rightarrow (S)S \mid \in$

$w = ()$

No.	Pila	Entrada	Acción
1	\$	()\$	desplazar
2	\$()\$	reducir $S \rightarrow \in$
3	\$(S)\$	desplazar
4	\$(S)	\$	reducir $S \rightarrow \in$
5	\$(S)S	\$	reducir $S \rightarrow (S)S$

$$S \Rightarrow (S)S \Rightarrow (S) \Rightarrow ()$$

Ejemplo 1

Dada la **GIC**: $S \rightarrow (S)S \mid \in$

La GIC aumentada es:

$S' \rightarrow S$

$S \rightarrow (S)S \mid \in$

$w = ()$

No.	Pila	Entrada	Acción
1	\$)\$	desplazar
2	\$()\$	reducir $S \rightarrow \in$
3	\$(S)\$	desplazar
4	\$(S)	\$	reducir $S \rightarrow \in$
5	\$(S)S	\$	reducir $S \rightarrow (S)S$
6	\$S	\$	reducir $S' \rightarrow S$

$S' \Rightarrow S \Rightarrow (S)S \Rightarrow (S) \Rightarrow ()$

Ejemplo 1

Dada la **GIC**: $S \rightarrow (S)S \mid \in$

La GIC aumentada es:

$S' \rightarrow S$

$S \rightarrow (S)S \mid \in$

$w = ()$

No.	Pila	Entrada	Acción
1	\$)\$	desplazar
2	\$())\$	reducir $S \rightarrow \in$
3	\$(\$)\$	desplazar
4	\$(\$)	\$	reducir $S \rightarrow \in$
5	\$(\$)S	\$	reducir $S \rightarrow (S)S$
6	\$S	\$	reducir $S' \rightarrow S$
7	\$S'	\$	aceptar

$S' \Rightarrow S \Rightarrow (S)S \Rightarrow (S) \Rightarrow ()$

Ejemplo 2

Dada la **GIC**: $E \rightarrow E+n \mid n$

La GIC aumentada es:

$E' \rightarrow E$

$E \rightarrow E+n \mid n$

$w=n+n$

No.	Pila	Entrada	Acción
1	\$	n+n\$	desplazar

$n+n$

Ejemplo 2

Dada la **GIC**: $E \rightarrow E+n \mid n$

La GIC aumentada es:

$E' \rightarrow E$

$E \rightarrow E+n \mid n$

$w=n+n$

No.	Pila	Entrada	Acción
1	\$	n+n\$	desplazar
2	\$n	+n\$	reducir $E \rightarrow n$

$E+n \Rightarrow n+n$

Ejemplo 2

Dada la **GIC**: $E \rightarrow E+n \mid n$

La GIC aumentada es:

$E' \rightarrow E$

$E \rightarrow E+n \mid n$

$w=n+n$

No.	Pila	Entrada	Acción
1	\$	n+n\$	desplazar
2	\$n	+n\$	reducir $E \rightarrow n$
3	\$E	+n\$	desplazar

$$E+n \xrightarrow{\hspace{1cm}} n+n$$

Ejemplo 2

Dada la **GIC**: $E \rightarrow E+n \mid n$

La GIC aumentada es:

$E' \rightarrow E$

$E \rightarrow E+n \mid n$

$w=n+n$

No.	Pila	Entrada	Acción
1	\$	n+n\$	desplazar
2	\$n	+n\$	reducir $E \rightarrow n$
3	\$E	+n\$	desplazar
4	\$E+	n\$	desplazar

$$E+n \xrightarrow{\quad} n+n$$

Ejemplo 2

Dada la **GIC**: $E \rightarrow E+n \mid n$

La GIC aumentada es:

$E' \rightarrow E$

$E \rightarrow E+n \mid n$

$w=n+n$

No.	Pila	Entrada	Acción
1	\$	n+n\$	desplazar
2	\$n	+n\$	reducir $E \rightarrow n$
3	\$E	+n\$	desplazar
4	\$E+	n\$	desplazar
5	\$E+n	\$	reducir $E \rightarrow E+n$

$$E \xrightarrow{\text{ }} E+n \xrightarrow{\text{ }} n+n$$

Ejemplo 2

Dada la **GIC**: $E \rightarrow E+n \mid n$

La GIC aumentada es:

$E' \rightarrow E$

$E \rightarrow E+n \mid n$

$w=n+n$

No.	Pila	Entrada	Acción
1	\$	n+n\$	desplazar
2	\$n	+n\$	reducir $E \rightarrow n$
3	\$E	+n\$	desplazar
4	\$E+	n\$	desplazar
5	\$E+n	\$	reducir $E \rightarrow E+n$
6	\$E	\$	reducir $E' \rightarrow E$

$E' \Rightarrow E \Rightarrow E+n \Rightarrow n+n$

Ejemplo 2

Dada la **GIC**: $E \rightarrow E+n \mid n$

La GIC aumentada es:

$E' \rightarrow E$

$E \rightarrow E+n \mid n$

$w=n+n$

No.	Pila	Entrada	Acción
1	\$	n+n\$	desplazar
2	\$n	+n\$	reducir $E \rightarrow n$
3	\$E	+n\$	desplazar
4	\$E+	n\$	desplazar
5	\$E+n	\$	reducir $E \rightarrow E+n$
6	\$E	\$	reducir $E' \rightarrow E$
7	\$E'	\$	aceptar

$E' \Rightarrow E \Rightarrow E+n \Rightarrow n+n$

Ejemplo 1 vs Ejemplo 2

GIC aumentada:

$$S' \rightarrow S$$

$$S \rightarrow (S)S \mid \epsilon$$

No.	Pila	Entrada	Acción
1	\$	()\$	desplazar
2	\$())\$	reducir $S \rightarrow \epsilon$
3	\$(S))\$	desplazar
4	\$(S)	\$	reducir $S \rightarrow \epsilon$
5	$\$(S)S$	\$	reducir $S \rightarrow (S)S$
6	$\$S$	\$	reducir $S' \rightarrow S$
7	$\$S'$	\$	aceptar

Línea 5: S está en la parte superior de la pila, y el ASA realiza una reducción $S \rightarrow (S)S$

Línea 6: S está en la parte superior de la pila , y el ASA realiza una reducción $S' \rightarrow S$

Un ASA puede necesitar examinar más allá de la parte superior de la pila para determinar la acción a ejecutar.

GIC aumentada:

$$E' \rightarrow E$$

$$E \rightarrow E+n \mid n$$

No.	Pila	Entrada	Acción
1	\$	$n+n\$$	desplazar
2	$\$n$	$+n\$$	reducir $E \rightarrow n$
3	$\$E$	$+n\$$	desplazar
4	$\$E+$	$n\$$	desplazar
5	$\$E+n$	\$	reducir $E \rightarrow E+n$
6	$\$E$	\$	reducir $E' \rightarrow E$
7	$\$E'$	\$	aceptar

Línea 3: E está en la parte superior de la pila, y el ASA realiza un desplazamiento. Terminal de entrada es +.

Línea 6: E está en la parte superior de la pila , y el ASA realiza una reducción $E' \rightarrow E$. Terminal a la entrada es \$.

Un ASA puede necesitar consultar el terminal en la entrada como una búsqueda hacia adelante. 21

Forma de Frase Más Derecha

Dada la **GIC**: $E \rightarrow E+n \mid n$

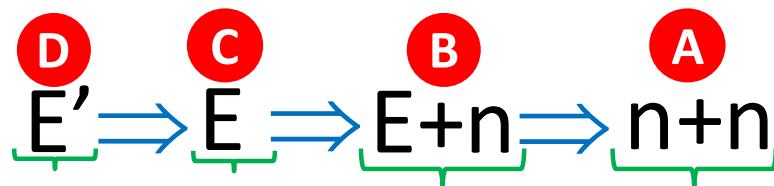
La GIC aumentada es:

$E' \rightarrow E$

$E \rightarrow E+n \mid n$

$w=n+n$

No.	Pila	Entrada	Acción
1	A \$	n+n\$	desplazar
2	A \$n	+n\$	reducir $E \rightarrow n$
3	B \$E	+n\$	desplazar
4	B \$E+	n\$	desplazar
5	B \$E+n	\$	reducir $E \rightarrow E+n$
6	C \$E	\$	reducir $E' \rightarrow E$
7	D \$E'	\$	aceptar



Formas de Frase Más Derecha

Cada forma de frase se divide entre la entrada y la pila durante un análisis sintáctico ascendente

Prefijo Viable

Dada la GIC: $E \rightarrow E+n \mid n$

La GIC aumentada es:

$E' \rightarrow E$

$E \rightarrow E+n \mid n$

$w=n+n$

Prefijos Viables de $E+n$

③ $E \parallel +n$

E

④ $E+ \parallel n$

$E+$

⑤ $E+n \parallel$

$E+n$

No.	Pila	Entrada	Acción
1	\$	$n+n\$$	desplazar
2	$\$n$	$+n\$$	reducir $E \rightarrow n$
3	$\$E$	$+n\$$	desplazar
4	$\$E+$	$n\$$	desplazar
5	$\$E+n$	\$	reducir $E \rightarrow E+n$
6	$\$E$	\$	reducir $E' \rightarrow E$
7	$\$E'$	\$	aceptar

Prefijo Viable

Se indica dónde está la parte superior de la pila en cada momento mediante el símbolo \parallel (división entre la pila y la entrada)

Prefijo Viable

Dada la GIC: $E \rightarrow E+n \mid n$

La GIC aumentada es:

$E' \rightarrow E$

$E \rightarrow E+n \mid n$

$w=n+n$

Prefijos Viables de $n+n$

① $\|n+n$

\in

② $n\|+n$

n

No.	Pila	Entrada	Acción
1	\$	$n+n\$$	desplazar
2	$\$n$	$+n\$$	reducir $E \rightarrow n$
3	$\$E$	$+n\$$	desplazar
4	$\$E+$	$n\$$	desplazar
5	$\$E+n$	\$	reducir $E \rightarrow E+n$
6	$\$E$	\$	reducir $E' \rightarrow E$
7	$\$E'$	\$	aceptar

Prefijo que lleva a una forma de fase

Prefijo Viable

Se indica dónde está la parte superior de la pila en cada momento mediante el símbolo $\|$ (división entre la pila y la entrada)

Autómatas Finitos de Elementos LR(0)

- **Elemento LR(0):** o simplemente Elemento de una GIC es una producción con una posición distinguida en su lado derecho.
- **Posición distinguida:** Se indica mediante un punto (metasímbolo, no es un terminal).
- **Ejemplo:** Si $A \rightarrow \alpha\beta$, luego $A \rightarrow \alpha.\beta$ es un elemento LR(0).

• ~~elemento canonico~~
~~símbolo gramatical~~

$A \xrightarrow{.} \alpha.\beta$

$A \xrightarrow{.} \alpha.\beta$

$A \xrightarrow{.} \alpha.\beta$

$A \xrightarrow{.} \alpha.\beta$

Por cada n símbolos gramaticales
Hay $n+1$ elementos canonicos

Elementos de GIC del Ejemplo 1

La GIC aumentada del Ejemplo 1:

$S' \rightarrow S$

$S \rightarrow (S)S \mid \in$

Tiene tres producciones y sus elementos son:

$S' \rightarrow .S$

$S' \rightarrow S.$

$S \rightarrow .(S)S$

$S \rightarrow (.S)S$

$S \rightarrow (S.)S$

$S \rightarrow (S).S$

$S \rightarrow (S)S.$

$S \rightarrow .$

Elementos de GIC del Ejemplo 2

La GIC aumentada del Ejemplo 2:

$$E' \rightarrow E$$

$$E \rightarrow E+n \mid n$$

Tiene tres producciones y sus elementos son:

$$E' \rightarrow .E$$

$$E' \rightarrow E.$$

$$E \rightarrow .E+n$$

$$E \rightarrow E.+n$$

$$E \rightarrow E+.n$$

$$E \rightarrow E+n.$$

$$E \rightarrow .n$$

$$E \rightarrow n.$$

Autómatas Finitos de Elementos

LR(0)

- **Idea General:** El elemento es un paso intermedio en el reconocimiento del lado derecho de una opción de producción.
- En el elemento $A \rightarrow \alpha.\beta$ construido de la producción $A \rightarrow \alpha\beta$, significa que
 - ya se ha analizado α .
 - y que se pueden derivar los siguientes terminales de la entrada a partir de β .
 - En términos de la pila de análisis sintáctico, esto significa que β debe aparecer en la cima de la pila.

Autómatas Finitos de Elementos LR(0)

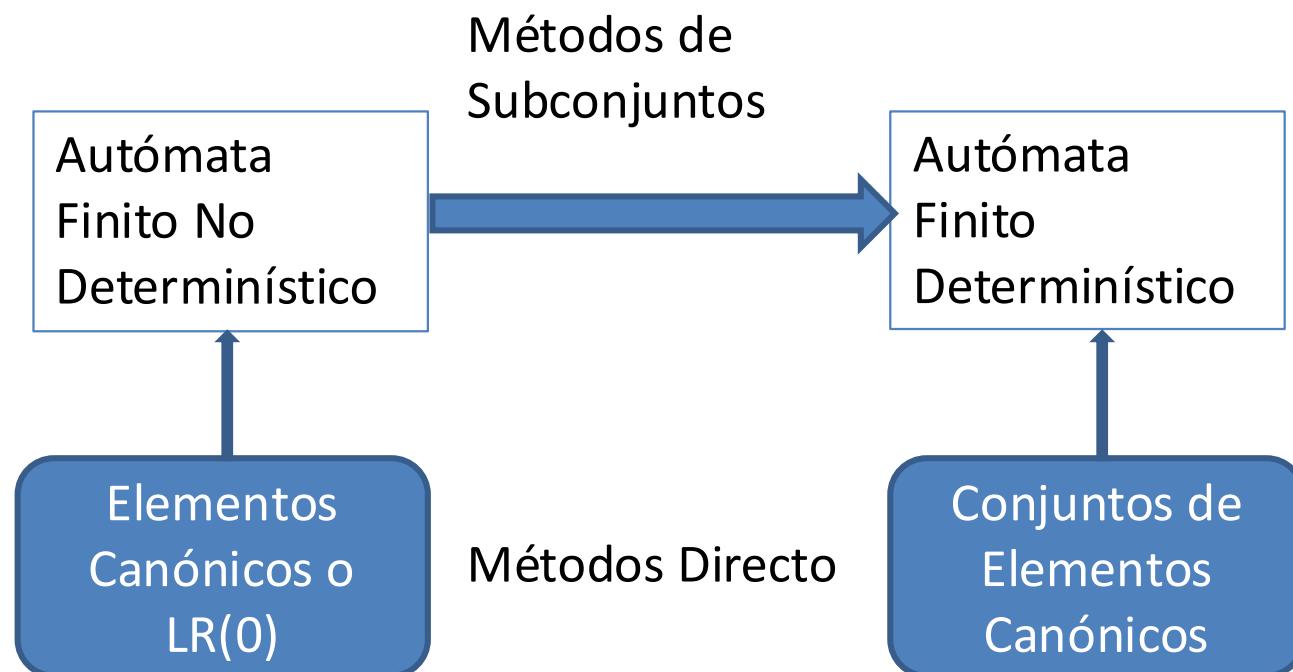
- Un elemento $A \rightarrow .\alpha$ significa que estamos cerca de reconocer una A mediante la selección de la producción $A \rightarrow \alpha$ (**Elementos iniciales**).
- Un elemento $A \rightarrow \alpha.$ significa que α está ubicado en la parte superior de la pila y puede ser un **mango**, si lo vamos a utilizar para la siguiente reducción (**Elementos completos**).

No determinístico

Autómatas Finitos de Elementos

LR(0)

- Los elementos LR(0) se pueden emplear como los estados de un A.F., los cuales mantienen información acerca de la pila y del progreso de un análisis por desplazamiento/reducción.

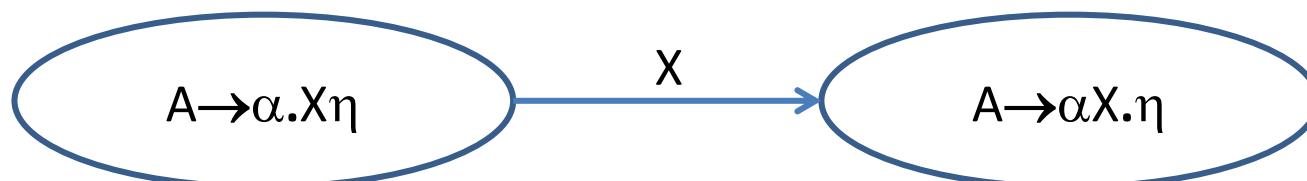


Autómatas Finitos de Elementos LR(0)

Transiciones:

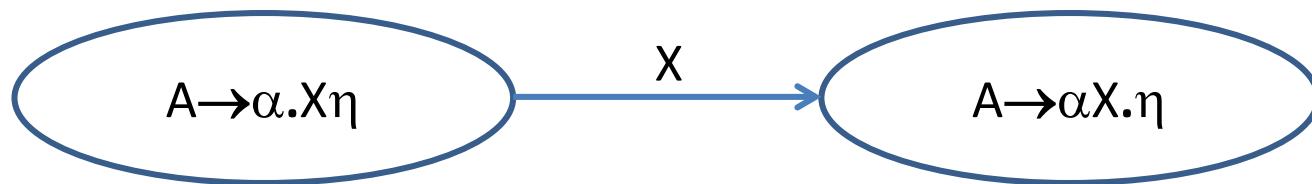
1. Si se tiene un elemento $A \rightarrow \alpha.X\eta$. Luego este elemento representa un estado del AFN.
2. X puede ser un terminal o un No terminal.
3. Dado que existe otro elemento $A \rightarrow \alpha X.\underline{\eta}$ y, por tanto representa otro estado.
4. Luego existe una transición del estado $A \rightarrow \alpha X\eta$ al estado $A \rightarrow \alpha X.\eta$ con el símbolo X.

Secuencia
de símbolos
gramaticales



Autómatas Finitos de Elementos LR(0)

Transiciones:



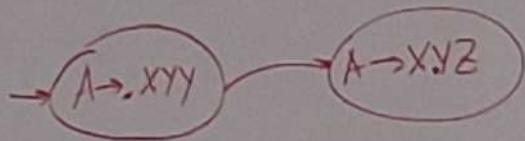
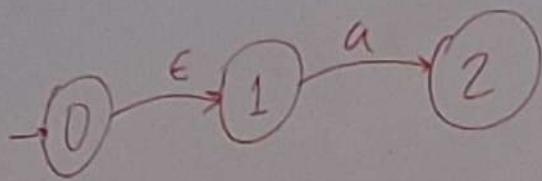
X es un terminal. Por qué se presenta una transición de esta forma?

Porque hubo un desplazamiento del terminal X desde la entrada hacia la pila

X es un No terminal. Por qué se presenta una transición de esta forma?

Porque hubo una reducción. Sin embargo X no estará como símbolo a la entrada.

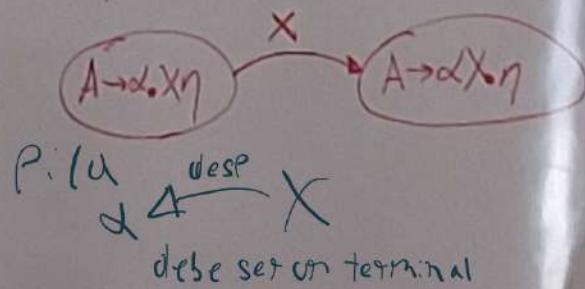
Qué procedimiento se sigue para que aparezca X en la cima de la pila?



$A \rightarrow \alpha \cdot X\eta$

1) $X \cdot T$
Despl

2) $X \cdot N$
Red



$P \vdash w$
 \downarrow Reduction

R14
\$
\$..a
\$..Y
\$..Yb
\$..X

Entrada
ab...\$
b...\$
b...\$
...\$
...\$

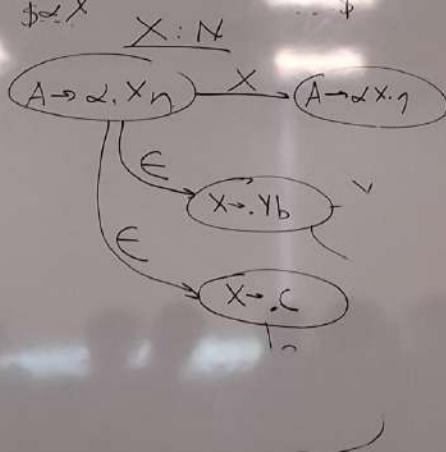
Acción
Despl
Red Y→a
Despl
Red X→Yb

X→Yb|C
Y→a

R14
\$
\$..a
\$..Y
\$..Yb
\$..X

Entrada
ab...\$
b...\$
b...\$
...\$
...\$

Acción
Despl
Red Y→a
Despl
Red X→Yb



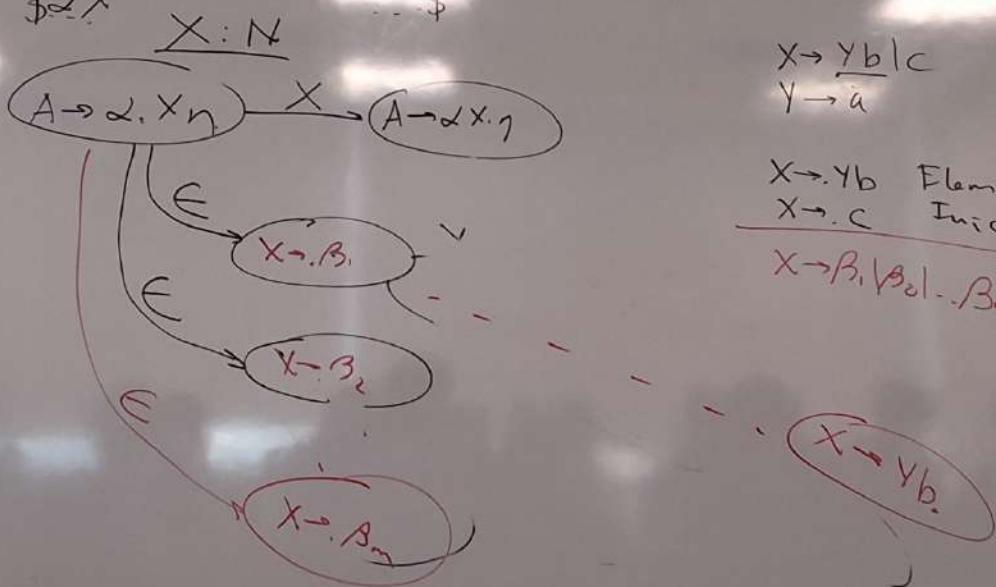
X→Yb|C
Y→a
X→Yb Elem
X→C Inic

HITACHI

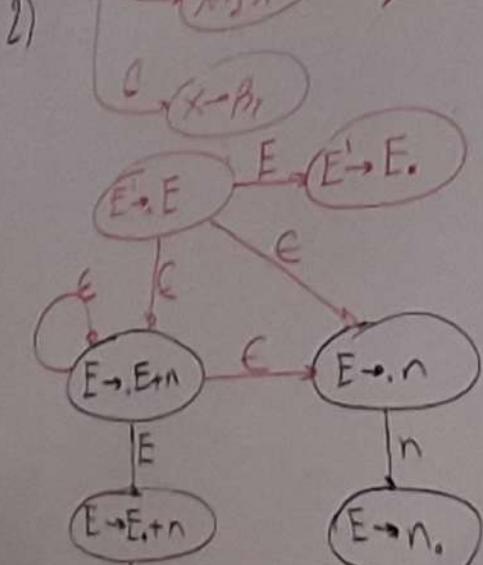
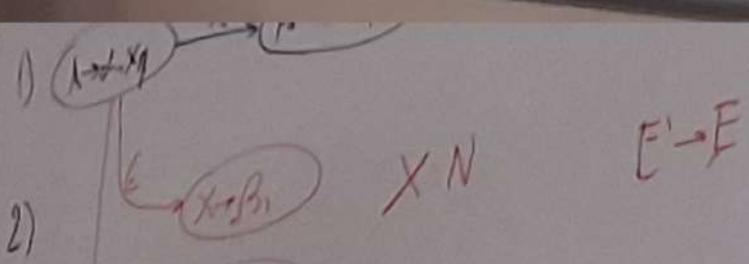
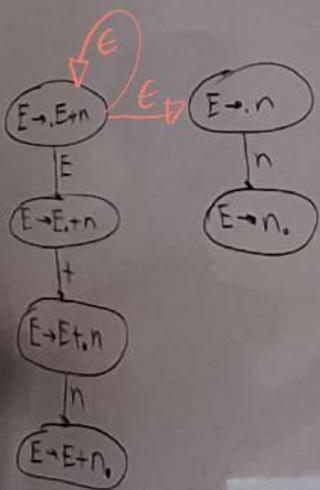
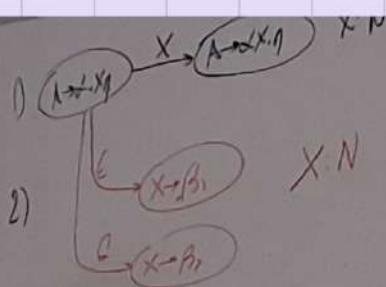
Fig.

Entrada
ab... \$
b... \$
b... \$
- \$
- \$

Acción
Despl.
Red. $Y \rightarrow a$
Despl.
Red. $X \rightarrow y_b$



La GIC del Ejemp
 $E \rightarrow E + n \mid n$

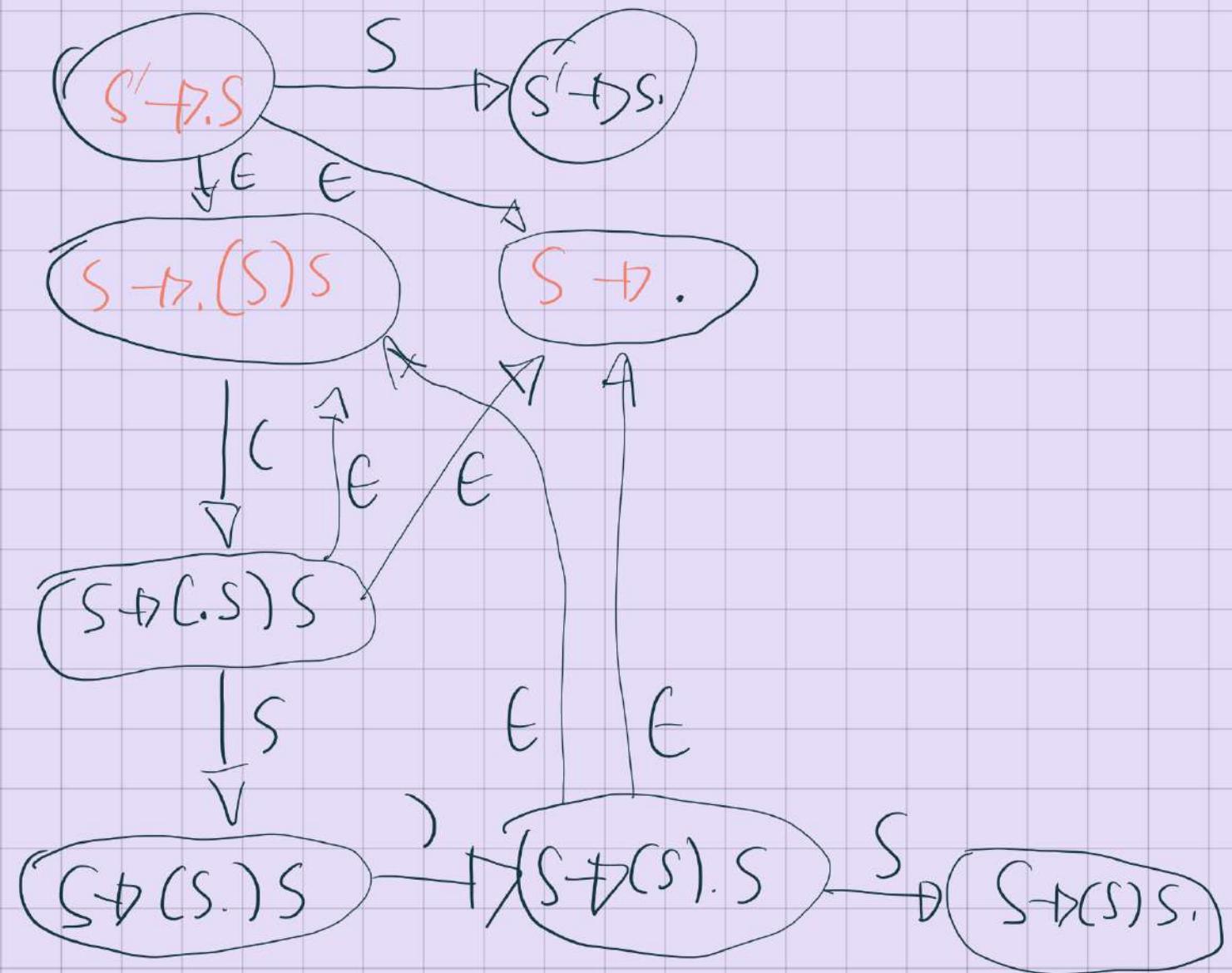


$E' \rightarrow E$

$\overline{x \rightarrow \beta_1}$

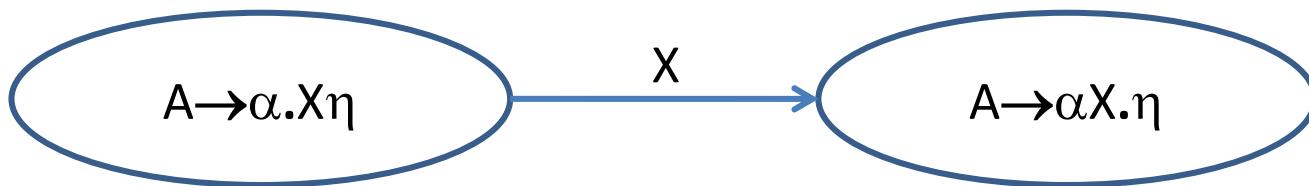
$\overline{x \rightarrow \beta_2}$

La
 $E -$



Autómatas Finitos de Elementos LR(0)

Transiciones:

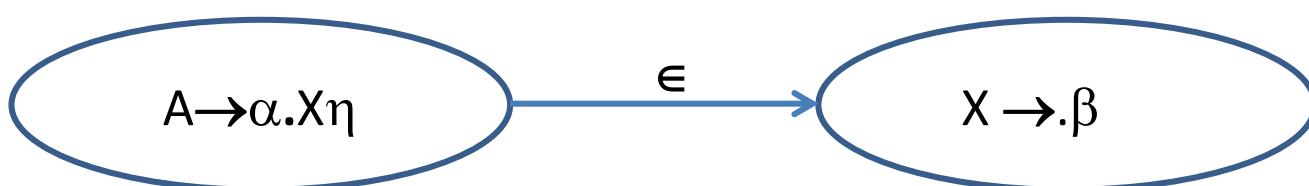


Cuando **X** es un **No terminal**, aparecerá en la cima de la **pila** durante una **reducción** donde interviene una producción $X \rightarrow \beta$.

Una **reducción** de esta clase debe estar precedida por el reconocimiento de una β . Es decir, β debe aparecer en la cima de la **pila**.

Luego el **estado** determinado por el **elemento** $X \rightarrow .\beta$ indica el comienzo de este proceso (el punto indica que estamos cerca de reconocer a β)

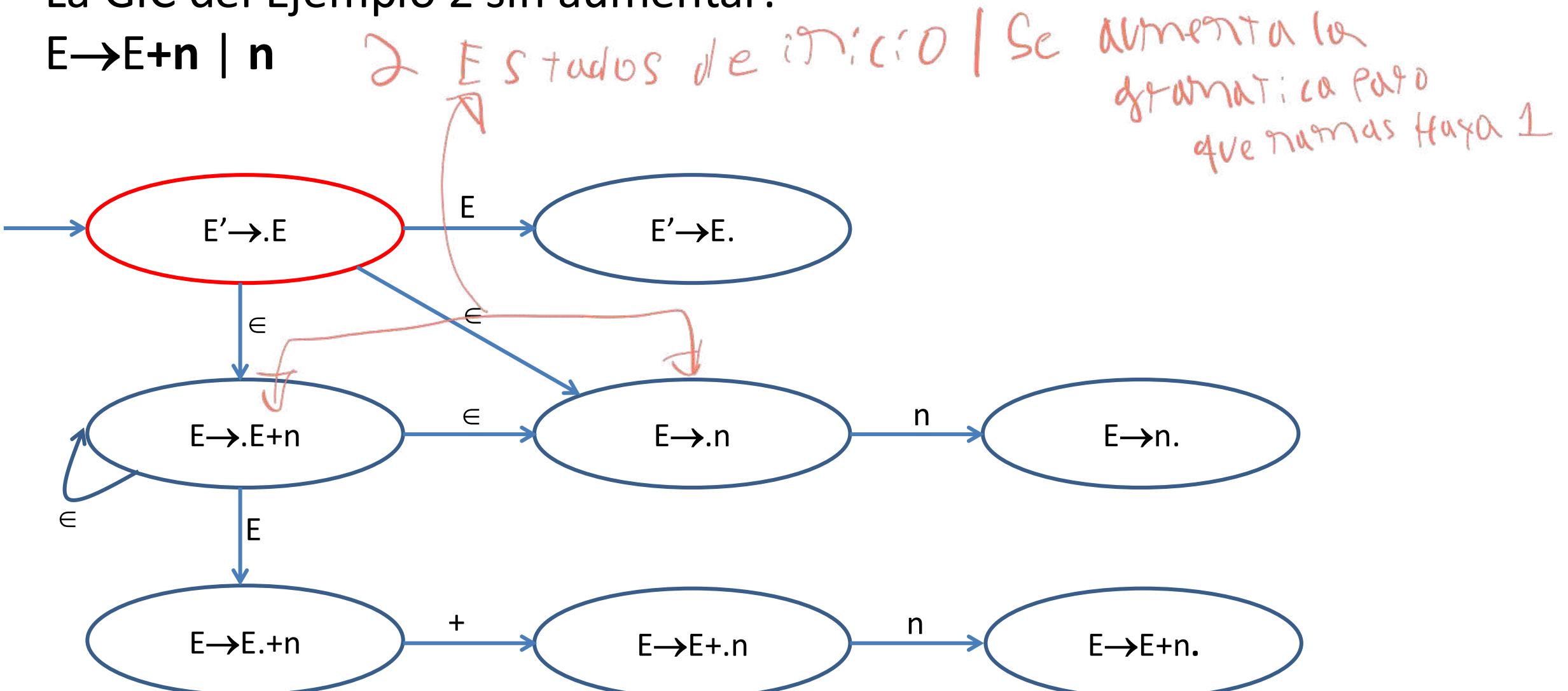
Por tanto, por cada **elemento** $A \rightarrow \alpha.X\eta$ se debe agregar una transición ϵ en cada producción $X \rightarrow \beta$ de X , indicando que se produce X mediante el reconocimiento de los lados derechos de sus producciones.



Ejemplo de AFN

La GIC del Ejemplo 2 sin aumentar:

$$E \rightarrow E + n \mid n$$



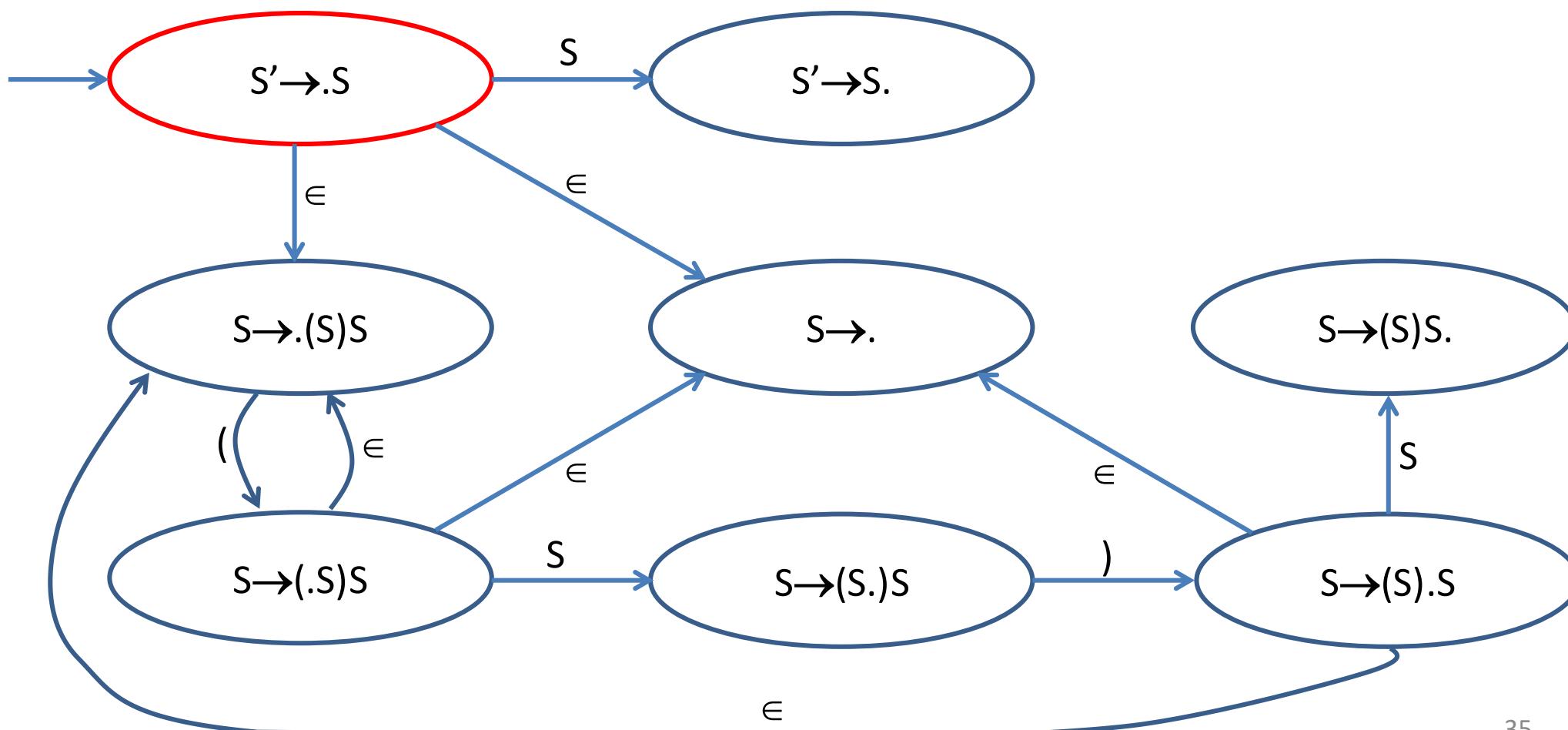
Ejemplo de AFN

La GIC del Ejemplo 1 aumentada:

$$S' \rightarrow S$$

$$S \rightarrow (S)S \mid \in$$

Construir el AFN para este ejemplo:



Autómatas Finitos de Elementos LR(0)

Los **Elementos Canónicos** ó **LR(0)** forman los **Conjuntos Canónicos**. Estos forman la **Colección Canónica**.

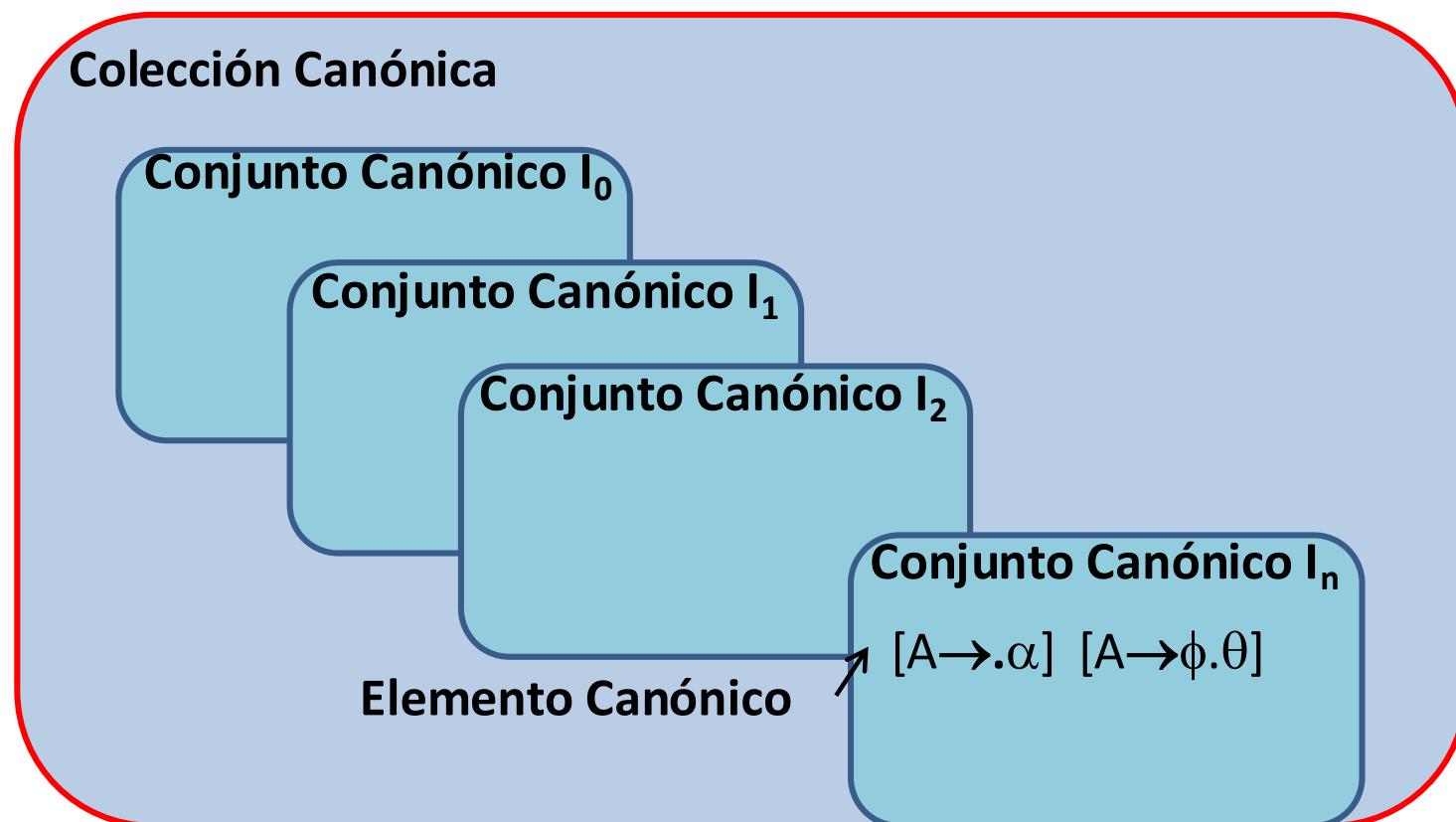
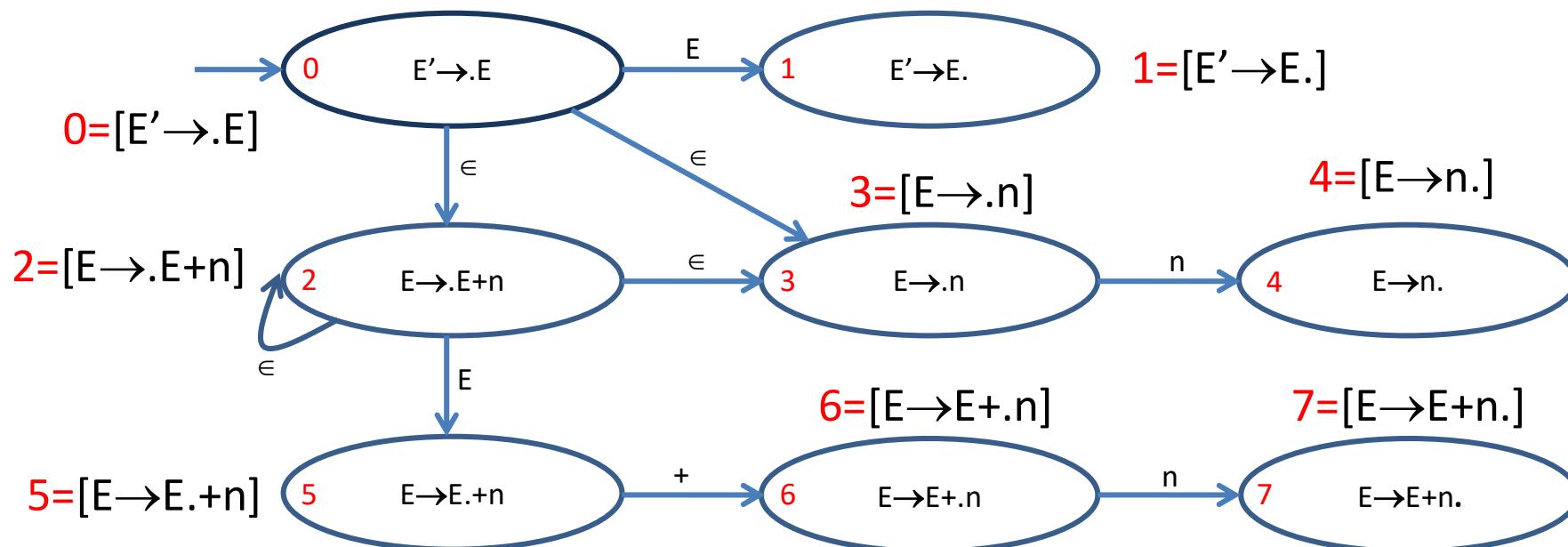


Tabla de Análisis Sintáctico SLR: AFD

Método de Subconjuntos



La GIC aumentada del Ejemplo 2:
 $E' \rightarrow E$
 $E \rightarrow E+n \mid n$

$1 = [E' \rightarrow E.]$

$4 = [E \rightarrow n.]$

$7 = [E \rightarrow E+n.]$

$$T = \text{Cerradura} - \in(\{0\}) = \{0, 2, 3\} = \text{Cerradura}(\{[E' \rightarrow .E]\}) = \{ [E' \rightarrow .E], [E \rightarrow .E+n], [E \rightarrow .n] \}$$

Cerradura(I): Si I es un conjunto de elementos para una GIC G , entonces $\text{Cerradura}(I)$ es el conjunto de elementos construido a partir de I por las reglas:

1. Inicialmente, todo elemento de I se añade a $\text{cerradura}(I)$.
2. Si $A \rightarrow \alpha.B\beta$ está en $\text{cerradura}(I)$ y $B \rightarrow \gamma$ es una producción, entonces añádase el elemento $B \rightarrow \gamma$ a $\text{cerradura}(I)$.

Ejemplo de Aplicación de *cerradura*

Dada la GIC aumentada:

$$\begin{array}{l} E' \rightarrow E \\ E \rightarrow E + T \mid T \\ T \rightarrow T^* F \mid F \\ F \rightarrow (E) \mid id \end{array}$$

cerradura(I)

$$\begin{array}{l} [E' \rightarrow .E] \\ [E \rightarrow .E + T] \\ [E \rightarrow .T] \\ [T \rightarrow .T^* F] \\ [T \rightarrow .F] \\ [F \rightarrow .(E)] \\ [F \rightarrow .id] \end{array}$$

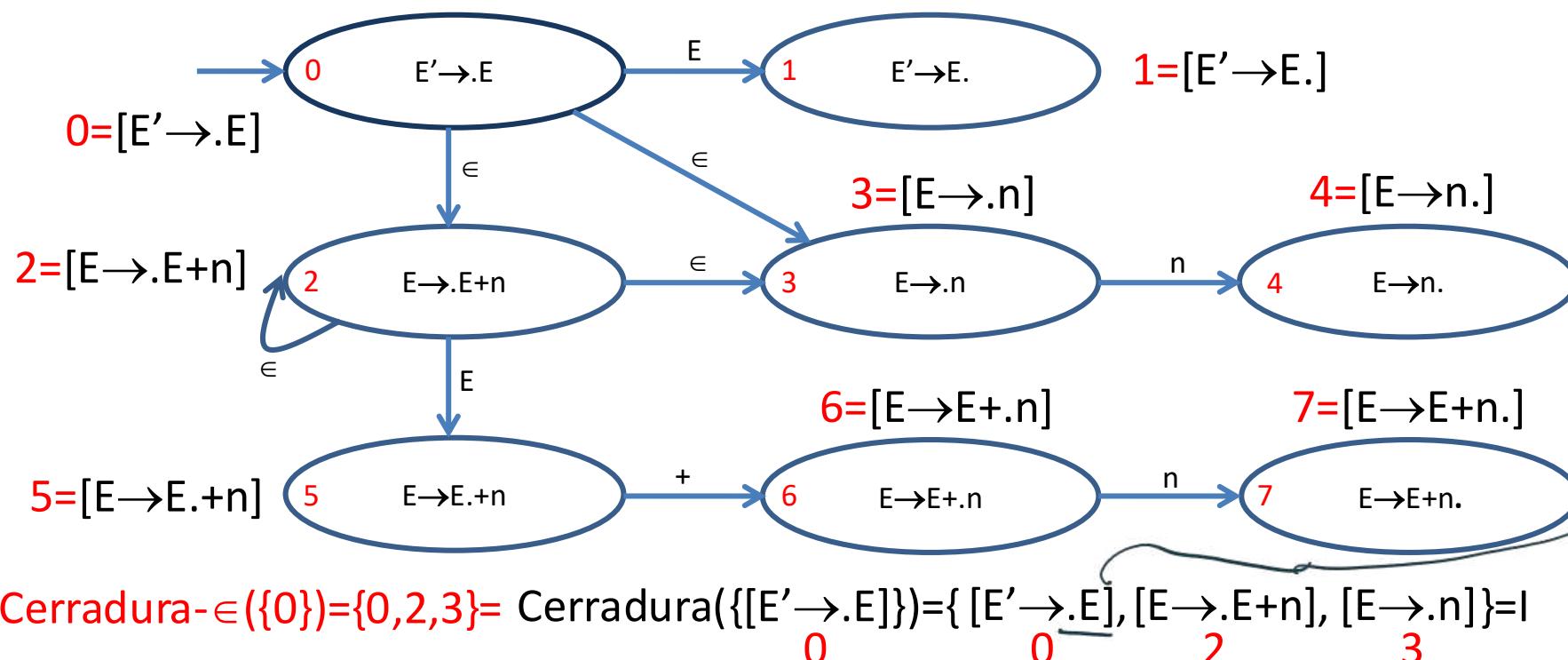
Si I es el conjunto de un elemento:

$$I = \{[E' \rightarrow .E]\}$$

```
function cerradura ( I );
begin
  J := I;
  repeat
    for cada elemento  $A \rightarrow \alpha \cdot B \beta$  en  $J$  y cada producción
       $B \rightarrow \gamma$  de  $G$  tal que  $B \rightarrow \cdot \gamma$  no esté en  $J$  do
        añadir  $B \rightarrow \cdot \gamma$  a  $J$ 
  until no se puedan añadir más elementos a  $J$ ;
  return J
end
```


Tabla de Análisis Sintáctico SLR: AFD

Método de Subconjuntos



La GIC aumentada del Ejemplo 2:

$$E' \rightarrow E$$

$$E \rightarrow E+n \mid n$$

$$1 = [E' \rightarrow E.]$$

$$2 = [E \rightarrow .E+n]$$

$$3 = [E \rightarrow .n]$$

$$4 = [E \rightarrow n.]$$

$$5 = [E \rightarrow E.+n]$$

$$6 = [E \rightarrow E+.n]$$

$$7 = [E \rightarrow E+n.]$$

$$T = \text{Cerradura} - \in(\{0\}) = \{0, 2, 3\} = \text{Cerradura}(\{[E' \rightarrow .E]\}) = \{ [E' \rightarrow .E], [E \rightarrow .E+n], [E \rightarrow .n] \} = \{ \underset{0}{[E' \rightarrow .E]}, \underset{2}{[E \rightarrow .E+n]}, \underset{3}{[E \rightarrow .n]} \}$$

$$\text{Cerradura}(\text{mueve}(T, E)) = \text{Cerradura}(\{1, 5\}) = \text{Cerradura}(\{ [E' \rightarrow E.], [E \rightarrow E.+n] \}) = \{ \underset{1}{[E' \rightarrow E.]}, \underset{5}{[E \rightarrow E.+n]} \}$$

$$\text{Cerradura}(\text{mueve}(T, n)) = \text{Cerradura}(\{4\}) = \text{Cerradura}(\{ [E \rightarrow n.] \}) = \{ \underset{4}{[E \rightarrow n.]} \}$$

$$\text{ir_a}(I, E) = \text{Cerradura}(\{ [E' \rightarrow E.], [E \rightarrow E.+n] \}) = \{ [E' \rightarrow E.], [E \rightarrow E.+n] \}$$

$\text{ir_a}(I, X)$: donde I es el conjunto de elementos y X es un símbolo gramatical. Se define $\text{ir_a}(I, X)$ como la cerradura del conjunto de todos los elementos $[A \rightarrow \alpha X \beta]$ tales que $[A \rightarrow \alpha X \beta]$ esté en I

Si el punto precede a un no terminal se agregan sus producciones precedidas de A

A → β₁ / β₂ se agregan {·β₁ / ·β₂}

$A \rightarrow \beta_1 / \beta_2$
se agregan $\{ \cdot \beta_1 / \cdot \beta_2 \}$

Ejemplo de Aplicación de *ir_a*

Dada la GIC aumentada:

$E' \rightarrow E$
 $E \rightarrow E + T \mid T$
 $T \rightarrow T^* F \mid F$
 $F \rightarrow (E) \mid id$

Si I es el conjunto de elementos:

$I = \{[E' \rightarrow E.], [E \rightarrow E.+T]\}$

$ir_a(I, +) = cerradura(\{[E \rightarrow E.+T]\})$

$[E \rightarrow E+.T]$
 $[T \rightarrow .T^*F]$
 $[T \rightarrow .F]$
 $[F \rightarrow .(E)]$
 $[F \rightarrow . id]$

Tabla de Análisis Sintáctico SLR: AFD

- ***Cerradura(I)***: Si I es un conjunto de elementos para una GIC G , entonces es el conjunto de elementos construido a partir de I por las reglas:
 1. Inicialmente, todo elemento de I se añade a $\text{cerradura}(I)$.
 2. Si $A \rightarrow \alpha.B\beta$ está en $\text{cerradura}(I)$ y $B \rightarrow \gamma$ es una producción, entonces añádase el elemento $B \rightarrow .\gamma$ a $\text{cerradura}(I)$.
- ***ir_a(I,X)***: donde I es el conjunto de elementos y X es un símbolo gramatical. Se define $\text{ir}_a(I,X)$ como la cerradura del conjunto de todos los elementos $[A \rightarrow \alpha X . \beta]$ tales que $[A \rightarrow \alpha X \beta]$ esté en I

Construcción de Conjuntos Canónicos

```
procedure elementos ( $G'$ );
begin
     $C := \{cerradura (\{[S' \rightarrow \cdot S]\})\};$ 
repeat
    for cada conjunto de elementos  $I$  en  $C$  y cada símbolo
        gramatical  $X$  tal que  $ir\_a (I, X)$  no esté vacío y no
        esté en  $C$  do
        añadir  $ir\_a (I, X)$  a  $C$ 
until no se puedan añadir más conjuntos de elementos a  $C$ 
end
```

Elemento inicial
↑
 A

ANÁLISIS SINTÁCTICO SLR

EJEMPLO

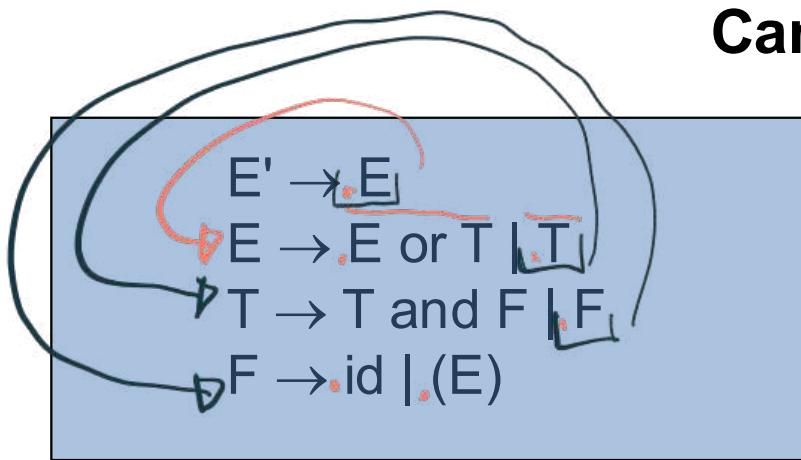
Gramática

$$\begin{aligned} E &\rightarrow E \text{ or } T \mid T \\ T &\rightarrow T \text{ and } F \mid F \\ F &\rightarrow id \mid (E) \end{aligned}$$

Gramática Aumentada

$$\begin{aligned} E' &\rightarrow E \\ E &\rightarrow E \text{ or } T \mid T \\ T &\rightarrow T \text{ and } F \mid F \\ F &\rightarrow id \mid (E) \end{aligned}$$

Conjuntos de Elementos Canónicos.



Cerradura($\{[E' \rightarrow .E]\}$)

= $\{[E' \rightarrow .E], [E \rightarrow .E \text{ or } T], [E \rightarrow .T], [T \rightarrow .T \text{ and } F],$

$[T \rightarrow .F], [F \rightarrow .id], [F \rightarrow .(E)]\}$

= \bullet_0 *conjunto canónico 1*

Se agregarán todos las producciones de E precedidas de un “.”

- Cerradura(I):** Si I es un conjunto de elementos para una GIC G , entonces es el conjunto de elementos construido a partir de I por las reglas:
1. Inicialmente, todo elemento de I se añade a $\text{cerradura}(I)$.
 2. Si $A \rightarrow \alpha.B\beta$ está en $\text{cerradura}(I)$ y $B \rightarrow \gamma$ es una producción, entonces añádase el elemento $B \rightarrow .\gamma$ a $\text{cerradura}(I)$.

Conjuntos de Elementos Canónicos.

$E' \rightarrow E$
 $E \rightarrow E \text{ or } T \mid T$
 $T \rightarrow T \text{ and } F \mid F$
 $F \rightarrow \text{id} \mid (E)$

$I_0 = \{[E' \rightarrow E], [E \rightarrow E \text{ or } T], [E \rightarrow .T], [T \rightarrow .T \text{ and } F], [T \rightarrow .F], [F \rightarrow .\text{id}], [F \rightarrow .(E)]\}$

$ir_a(I, X)$: donde I es el conjunto de elementos y X es un símbolo gramatical. Se define $ir_a(I, X)$ como la cerradura del conjunto de todos los elementos $[A \rightarrow \alpha X \beta]$ tales que $[A \rightarrow \alpha X \beta]$ esté en I

$$\begin{aligned}
 & Ir_a(I_0, E) \\
 &= \text{Cerradura}(\{[E' \rightarrow E.], [E \rightarrow E. \text{ or } T]\}) \\
 &= \{[E' \rightarrow E.], [E \rightarrow E. \text{ or } T]\} \quad \text{ambos tienen .E} \\
 &= I_1
 \end{aligned}$$

ambos tienen .E

$$\begin{aligned}
 & Ir_a(I_0, T) \\
 &= \text{Cerradura}(\{[E \rightarrow T.], [T \rightarrow T. \text{ and } F]\}) \\
 &= \{[E \rightarrow T.], [T \rightarrow T. \text{ and } F]\} \quad \{ \} \\
 &= I_2
 \end{aligned}$$

tienen .T

Cerradura(I): Si I es un conjunto de elementos para una GIC G , entonces es el conjunto de elementos construido a partir de I por las reglas:

1. Inicialmente, todo elemento de I se añade a $\text{cerradura}(I)$.
2. Si $A \rightarrow \alpha \cdot B \beta$ está en $\text{cerradura}(I)$ y $B \rightarrow \gamma$ es una producción, entonces añádase el elemento $B \rightarrow \cdot \gamma$ a $\text{cerradura}(I)$.

No se agrega porque el punto
 no precede a un
 no terminal



Conjuntos de Elementos Canónicos.

$E' \rightarrow E$
 $E \rightarrow E \text{ or } T \mid T$
 $T \rightarrow T \text{ and } F \mid F$
 $F \rightarrow id \mid (E)$

$I_0 = \{[E' \rightarrow .E], [E \rightarrow _E \text{ or } T],$
 $[E \rightarrow _T], [T \rightarrow .T \text{ and } F],$
 $[T \rightarrow .F], [F \rightarrow .id], [F \rightarrow .(E)]\}$

Ir_a(I_0, F)

$$\begin{aligned}
 &= \text{Cerradura}(\{[T \rightarrow F.]]) \\
 &= \{[T \rightarrow F.] \text{ } \text{no precede a } \text{ } \text{in } A\} \\
 &= I_3
 \end{aligned}$$

Ir_a(I_0, id)

$$\begin{aligned}
 &= \text{Cerradura}(\{[F \rightarrow id.]]) \\
 &= \{[F \rightarrow id.] \} \\
 &= I_4
 \end{aligned}$$

Ir_a($I_0, ()$)

$$\begin{aligned}
 &= \text{Cerradura}(\{[F \rightarrow (.E)]\}) \\
 &= \{[F \rightarrow (.E)], [E \rightarrow .E \text{ or } T], [E \rightarrow .T], [T \rightarrow .T \text{ and } F], [T \rightarrow .F], \\
 &[F \rightarrow .id], [F \rightarrow .(E)]\} \\
 &= I_5
 \end{aligned}$$

Cerradura(I): Si I es un conjunto de elementos para una GIC G , entonces es el conjunto

de elementos construido a partir de I por las reglas:

1. Inicialmente, todo elemento de I se añade a *cerradura(I)*.
2. Si $A \rightarrow \alpha.B\beta$ está en *cerradura(I)* y $B \rightarrow \gamma$ es una producción, entonces añádase el elemento $B \rightarrow .\gamma$ a *cerradura(I)*.

ir_a(I, X): donde I es el conjunto de elementos y X es un símbolo gramatical. Se define *ir_a(I, X)* como la cerradura del conjunto de todos los elementos $[A \rightarrow \alpha X \beta]$ tales que $[A \rightarrow \alpha X \beta]$ esté en I

Se agregan las producciones recursivas de E



Conjuntos de Elementos Canónicos.

$E' \rightarrow E$
 $E \rightarrow E \text{ or } T \mid T$
 $T \rightarrow T \text{ and } F \mid F$
 $F \rightarrow \text{id} \mid (E)$

$I_1 = \{[E' \rightarrow E.], [E \rightarrow E.\text{or } T]\}$

\cancel{X} *el X puede ser
lo que sea que este
precedido por .*

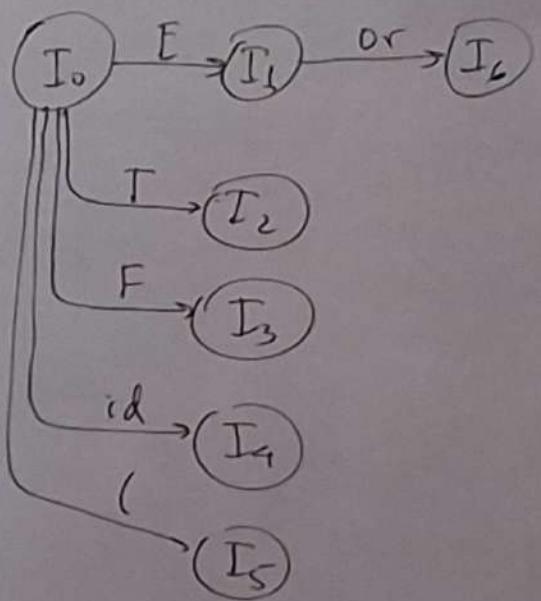
$$\begin{aligned} Ir_a(I_1, \text{or}) &= \text{Cerradura}(\{[E \rightarrow E \text{ or. } T]\}) \quad \text{excepto vacío} \\ &= \{[E \rightarrow E \text{ or. } T], [T \rightarrow .T \text{ and } F], [T \rightarrow .F], [F \rightarrow .\text{id}], [F \rightarrow .(E)]\} \\ &= I_6 \end{aligned}$$

Cerradura(I): Si I es un conjunto de elementos para una GIC G , entonces es el conjunto

de elementos construido a partir de I por las reglas:

1. Inicialmente, todo elemento de I se añade a *cerradura(I)*.
2. Si $A \rightarrow \alpha.B\beta$ está en *cerradura(I)* y $B \rightarrow \gamma$ es una producción, entonces añádase el elemento $B \rightarrow \gamma$ a *cerradura(I)*.

ir_a(I, X): donde I es el conjunto de elementos y X es un símbolo gramatical. Se define $ir_a(I, X)$ como la cerradura del conjunto de todos los elementos $[A \rightarrow \alpha X \beta]$ tales que $[A \rightarrow \alpha X \beta]$ esté en I



Conjuntos de Elementos Canónicos.

```
E' → E  
E → E or T | T  
T → T and F | F  
F → id | (E)
```

$I_2 = \{[E \rightarrow T], [T \rightarrow T \text{ and } F]\}$

$$\begin{aligned} Ir_a(I_2, \text{and}) \\ = & \quad \text{Cerradura}(\{[T \rightarrow T \text{ and. } F]\}) \\ = & \quad \{[T \rightarrow T \text{ and. } F], [F \rightarrow .\text{id}], [F \rightarrow .(E)]\} \\ = & \quad I_7 \end{aligned}$$

Cerradura(I): Si I es un conjunto de elementos para una GIC G , entonces es el conjunto

de elementos construido a partir de I por las reglas:

1. Inicialmente, todo elemento de I se añade a *cerradura*(I).
2. Si $A \rightarrow \alpha.B\beta$ está en *cerradura*(I) y $B \rightarrow \gamma$ es una producción, entonces añádase el elemento $B \rightarrow .\gamma$ a *cerradura*(I).

ir_a(I, X): donde I es el conjunto de elementos y X es un símbolo gramatical. Se define *ir_a(I, X)* como la cerradura del conjunto de todos los elementos $[A \rightarrow \alpha X \beta]$ tales que $[A \rightarrow \alpha X \beta]$ esté en I



Conjuntos de Elementos Canónicos.

$E' \rightarrow E$
 $E \rightarrow E \text{ or } T \mid T$
 $T \rightarrow T \text{ and } F \mid F$
 $F \rightarrow \text{id} \mid (E)$

$I_5 = \{ [F \rightarrow (.E)], [E \rightarrow .E \text{ or } T],$
 $[E \rightarrow .T], [T \rightarrow .T \text{ and } F],$
 $[T \rightarrow .F], [F \rightarrow .\text{id}], [F \rightarrow .(E)] \}$

$$\begin{aligned} & Ir_a(I_5, E) \\ &= \text{Cerradura}(\{[F \rightarrow (E.)], [E \rightarrow E. \text{ or } T]\}) \\ &= \{[F \rightarrow (E.)], [E \rightarrow E. \text{ or } T]\} \\ &= I_8 \end{aligned}$$

$$\begin{aligned} & Ir_a(I_5, T) \\ &= \text{Cerradura}(\{[E \rightarrow T.], [T \rightarrow T. \text{ and } F]\}) \\ &= I_2 \end{aligned}$$

Conjuntos de Elementos Canónicos.

$E' \rightarrow E$
 $E \rightarrow E \text{ or } T \mid T$
 $T \rightarrow T \text{ and } F \mid F$
 $F \rightarrow \text{id} \mid (E)$

$I_5 = \{[F \rightarrow (.E)], [E \rightarrow .E \text{ or } T], [E \rightarrow .T], [T \rightarrow .T \text{ and } F], [T \rightarrow .F], [F \rightarrow .\text{id}], [F \rightarrow .(E)]\}$

$$\begin{aligned} & Ir_a(I_5, F) \\ &= \text{Cerradura}(\{[T \rightarrow F.]]) \\ &= I_3 \end{aligned}$$

$$\begin{aligned} & Ir_a(I_5, \text{id}) \\ &= \text{Cerradura}(\{[F \rightarrow \text{id}.]\}) \\ &= I_4 \end{aligned}$$

$$\begin{aligned} & Ir_a(I_5, ()) \\ &= \text{Cerradura}(\{[F \rightarrow (.E)]\}) \\ &= I_5 \end{aligned}$$

Conjuntos de Elementos Canónicos.

$E' \rightarrow E$
 $E \rightarrow E \text{ or } T \mid T$
 $T \rightarrow T \text{ and } F \mid F$
 $F \rightarrow \text{id} \mid (E)$

$I_6 = \{[E \rightarrow E \text{ or. } T], [T \rightarrow .T \text{ and } F], [T \rightarrow .F], [F \rightarrow .\text{id}], [F \rightarrow .(E)]\}$

$$\begin{aligned} Ir_a(I_6, T) &= \text{Cerradura}(\{[E \rightarrow E \text{ or } T.], [T \rightarrow T. \text{ and } F]\}) \\ &= \{[E \rightarrow E \text{ or } T.], [T \rightarrow T. \text{ and } F]\} \\ &= I_9 \end{aligned}$$

$$\begin{aligned} Ir_a(I_6, F) &= \text{Cerradura}(\{[T \rightarrow F.]\}) \\ &= I_3 \end{aligned}$$



Conjuntos de Elementos Canónicos.

$E' \rightarrow E$
 $E \rightarrow E \text{ or } T \mid T$
 $T \rightarrow T \text{ and } F \mid F$
 $F \rightarrow id \mid (E)$

$I_6 = \{[E \rightarrow E \text{ or. } T], [T \rightarrow .T \text{ and } F], [T \rightarrow .F], [F \rightarrow .id], [F \rightarrow .(E)]\}$

$$\begin{aligned} Ir_a(I_6, id) \\ = & \quad \text{Cerradura } (\{[F \rightarrow id.]]) \\ = & \quad I_4 \end{aligned}$$

$$\begin{aligned} Ir_a(I_6, ()) \\ = & \quad \text{Cerradura}(\{[F \rightarrow (.E)]\}) \\ = & \quad I_5 \end{aligned}$$

Conjuntos de Elementos Canónicos.

$E' \rightarrow E$
 $E \rightarrow E \text{ or } T \mid T$
 $T \rightarrow T \text{ and } F \mid F$
 $F \rightarrow \text{id} \mid (E)$

$I_7 = \{[T \rightarrow T \text{ and. } F], [F \rightarrow .\text{id}]$
 $[F \rightarrow .(E)]\}$

$$\begin{aligned} Ir_a(I_7, F) \\ = & \quad \text{Cerradura}(\{[T \rightarrow T \text{ and. } F]\}) \\ = & \quad \{[T \rightarrow T \text{ and. } F]\} \\ = & \quad \boxed{10} \end{aligned}$$

$$\begin{aligned} Ir_a(I_7, \text{id}) \\ = & \quad \text{Cerradura} (\{[F \rightarrow \text{id}.]\}) \\ = & \quad \boxed{4} \end{aligned}$$

$$\begin{aligned} Ir_a(I_7, ()) \\ = & \quad \text{Cerradura}(\{[F \rightarrow (.E)]\}) \\ = & \quad \boxed{5} \end{aligned}$$



Conjuntos de Elementos Canónicos.

$E' \rightarrow E$
 $E \rightarrow E \text{ or } T \mid T$
 $T \rightarrow T \text{ and } F \mid F$
 $F \rightarrow \text{id} \mid (E)$

$I_8 = \{[F \rightarrow (E)] \mid [E \rightarrow E \text{ or } T]\}$

$$\begin{aligned} & Ir_a(I_8,) \\ &= \text{Cerradura}(\{[F \rightarrow (E).]\}) \\ &= \{[F \rightarrow (E).]\} \\ &= I_{11} \end{aligned}$$

$$\begin{aligned} & Ir_a(I_8, \text{or}) \\ &= \text{Cerradura}(\{[E \rightarrow E \text{ or. } T]\}) \\ &= I_6 \end{aligned}$$



Conjuntos de Elementos Canónicos.

$E' \rightarrow E$
 $E \rightarrow E \text{ or } T \mid T$
 $T \rightarrow T \text{ and } F \mid F$
 $F \rightarrow \text{id} \mid (E)$

$I_9 = \{[E \rightarrow E \text{ or } T.],$
 $[T \rightarrow T. \text{and } F]\}$

$$\begin{aligned} Ir_a(I_9, \text{and}) \\ = & \quad \text{Cerradura}(\{[T \rightarrow T \text{ and. } F]\}) \\ = & \quad I_7 \end{aligned}$$

Tabla TrandD de Autómata Finito Determinístico

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Tabla con encabezados de *acción* e *ir a*

Tabla SLR Resultante

Importante
aggregat signo \$
↓

Tabla con encabezados de *acción* e *ir a*

Encabezados de sección *acción* :símbolos terminales y S

Encabezados de sección *ir a* : símbolos no terminales

Tabla SLR Resultante

Regla 2: El estado i se construye a partir del conjunto canónico I_i . Cada conjunto canónico I_i define el estado

Tabla SLR Resultante

Regla 2: El estado i se construye a partir del conjunto canónico I_i . Cada conjunto canónico I_i define el estado

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Regla 2.a): Si $[A \rightarrow a.a\beta]$ está en I_i e $ir_a(I_i, a) = I_j$, entonces
Asignar “desplazar j ” a $acción[i, a]$.

Ejemplo: [Diap 44-46](#), [Diap 45-47](#) y [Diap. 45-48](#)

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0										
1										
2										
3										
4										
5										
6										
7										
8										
9										
10										
11										61

$[F \rightarrow .id]$ en I_0 y $ir_a(I_0, id) = I_4$
acción(0, id) = desplazar 4

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅		I ₉	I ₃	
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Regla 2.a): Si $[A \rightarrow a.a\beta]$ está en I_i e $ir_a(I_i, a) = I_j$, entonces
Asignar “desplazar j ” a $acción[i, a]$.

Ejemplo: Diap 44-46, Diap 45-47 y Diap. 45-48

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4							
1										
2										
3										
4										
5										
6										
7										
8										
9										
10										
11										62

[F → .id] en I₀ y ir_a(I₀, id) = I₄
acción(0, id) = desplazar 4

[F → .(E)] en I₀ y ir_a(I₀, ()) = I₅
acción(0, ()) = desplazar 5

Tabla SLR Resultante

Regla 2.a): Si $[A \rightarrow \alpha. a\beta]$ está en I_i e $ir_a(I_i, a) = I_j$, entonces Asignar “*desplazar j*” a *acción* $[i, a]$.

Ejemplo: Diap 44-46, Diap 45-47 y Diap. 45-48

[F → .id] en I_0 y $Ir_a(I_0, id) = I_4$
acción(0,id)= desplazar 4

[F → .(E)] en I₀ y lr_a(I₀,())= I₅
acción(0,())= desplazar 5

[E → E.orT] en I_1 y $Ir_a(I_1, or) = I_6$
acción(1,or)= desplazar 6

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Regla 2.a): Si $[A \rightarrow a.a\beta]$ está en I_i e $ir_a(I_i, a) = I_j$, entonces
Asignar “desplazar j ” a $acción[i, a]$.

Ejemplo: Diap 44-46, Diap 45-47 y Diap. 45-48

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6									
2										
3										
4										
5										
6										
7										
8										
9										
10										
11										64

$[F \rightarrow .id]$ en I_0 y $ir_a(I_0, id) = I_4$
acción(0,id)= desplazar 4

$[F \rightarrow .(E)]$ en I_0 y $ir_a(I_0, ()) = I_5$
acción(0,())= desplazar 5

$[E \rightarrow E.orT]$ en I_1 y $ir_a(I_1, or) = I_6$
acción(1,or)= desplazar 6

$[T \rightarrow T.andF]$ en I_2 y $ir_a(I_2, and) = I_7$
acción(2, and)= desplazar 7

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Regla 2.a): Si $[A \rightarrow a.a\beta]$ está en I_i e $ir_a(I_i, a) = I_j$, entonces
Asignar “desplazar j ” a $acción[i, a]$.

Ejemplo: Diap 44-46, Diap 45-47 y Diap. 45-48

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6									
2		d7								
3										
4										
5										
6										
7										
8										
9										
10										
11										65

$[F \rightarrow .id]$ en I_0 y $ir_a(I_0, id) = I_4$
acción(0,id)= desplazar 4

$[F \rightarrow .(E)]$ en I_0 y $ir_a(I_0, ()) = I_5$
acción(0,())= desplazar 5

$[E \rightarrow E.orT]$ en I_1 y $ir_a(I_1, or) = I_6$
acción(1,or)= desplazar 6

$[T \rightarrow T.andF]$ en I_2 y $ir_a(I_2, and) = I_7$
acción(2, and)= desplazar 7

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Regla 2.a): Si $[A \rightarrow a.a\beta]$ está en I_i e $ir_a(I_i, a) = I_j$, entonces
Asignar “desplazar j ” a $acción[i, a]$.

Ejemplo: Diap 44-46, Diap 45-47 y Diap. 45-48

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6									
2		d7								
3										
4										
5										
6										
7										
8										
9										
10										
11										66

En el AFD (TranD) se observa a la izquierda de la **línea roja**, que todos los I_i se pueden convertir, en las mismas posiciones de la matriz *acción*, en **d_i** o desplazar i.

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅		I ₉	I ₃	
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Regla 2.a): Si $[A \rightarrow a.a\beta]$ está en I_i e $ir_a(I_i, a) = I_j$, entonces
Asignar “desplazar j ” a $acción[i, a]$.

Ejemplo: Diap 44-46, Diap 45-47 y Diap. 45-48

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6									
2		d7								
3										
4										
5			d4	d5						
6			d4	d5						
7			d4	d5						
8	d6					d11				
9		d7								
10										
11										67

En el AFD (TranD) se observa a la izquierda de la **línea roja**, que todos los I_i se pueden convertir, en las mismas posiciones de la matriz acción, en **d_i** o desplazar i.

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

- 1) E → E or T
- 2) E → T
- 3) T → T and F
- 4) T → F
- 5) F → id
- 6) F → (E)

[E → T.] en I₂

Sgte(E) = { \$, or,) }

acción(2,\$)=red E → T=r2

acción(2,or)=red E → T=r2

acción(2,))=red E → T=r2

Tabla SLR Resultante

Regla 2.b): Si [A → α.] está en I_i, entonces asignar “reducir A → α” a acción[i,a] para toda a en SIGUIENTE(A). A no puede ser S’.

Ejemplo: Diap. 45 (2 ejemplos).

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6									
2			d7							
3										
4										
5			d4	d5						
6			d4	d5						
7			d4	d5						
8	d6					d11				
9			d7							
10										
11										

$\rightarrow d_5$

conclu:

1) $[A \rightarrow a, a\beta]$ existe en T_i

2) $ir_a(T_i, a) \rightarrow T_j$

con a genera el conjunto
de elementos
concretos T_j

$$\begin{aligned} ir_a(I_0, F) &= \text{Cerradura}(\{[T \rightarrow F]\}) \\ &= \{[T \rightarrow F]\} \\ &= I_3 \end{aligned}$$

$$\begin{aligned} ir_a(I_0, id) &= \text{Cerradura}(\{[F \rightarrow id]\}) \\ &= \{[F \rightarrow id]\} \\ &= I_4 \end{aligned}$$

$$\begin{aligned} ir_a(I_0, ()) &= \text{Cerradura}(\{[F \rightarrow (.E)]\}) \\ &= \{[F \rightarrow (.E)], [E \rightarrow .E \text{ or } T], [E \rightarrow .T], [T \rightarrow .T \text{ and } F], [T \rightarrow .F], \\ &\quad [F \rightarrow .id], [F \rightarrow .(E)]\} \end{aligned}$$

definición de elementos construido a partir de I por las reglas:
 1. Inicialmente, todo elemento de I se añade a cerradura(I).
 2. Si $A \rightarrow a.B\beta$ está en cerradura(I) y $B \rightarrow \gamma$ es una
producción, entonces añádase el elemento $B \rightarrow \gamma$ a
cerradura(I).

$ir_a(I, X)$: donde I es el conjunto de
elementos y X es un símbolo gramatical.
Se define $ir_a(I, X)$ como la cerradura del
conjunto de todos los elementos
 $[A \rightarrow aX\beta]$ tales que $[A \rightarrow aX\beta]$ esté en I

$[F \rightarrow .id]$ está en I_0 $[F \rightarrow .(E)]$ está en I_0

$$ir_a(T_0, id) = I_4$$

$$ir_a(T_0, ()) = I_5$$

$$\text{accion}(0, id) = \underline{d_4}$$

$$\text{accion}(0, ()) = \underline{d_5}$$

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Regla 2.b): Si $[A \rightarrow \alpha.]$ está en I_i , entonces asignar “reducir $A \rightarrow \alpha$ ” a $\text{acción}[i, a]$ para toda a en $\text{SIGUIENTE}(A)$. A no puede ser S ’.

Ejemplo: Diap. 45 (2 ejemplos).

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow \text{id}$
- 6) $F \rightarrow (E)$

$[E \rightarrow T.]$ en I_2

$\text{Sgte}(E) = \{\$, \text{or},)\}$

$\text{acción}(2, \$) = \text{red } E \rightarrow T = r2$

$\text{acción}(2, \text{or}) = \text{red } E \rightarrow T = r2$

$\text{acción}(2,)) = \text{red } E \rightarrow T = r2$

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6									
2	r2	d7			r2	r2				
3										
4										
5			d4	d5						
6			d4	d5						
7			d4	d5						
8	d6				d11					
9		d7								
10										
11										

Cond:

1) $[A \rightarrow 2.A/B]$ en I_i
2) $\text{ir_a}(I_i, a) = I_j$

\Rightarrow
 $\text{acción}[i, a] = \text{desp } j$

1) $[F \rightarrow .(d)]$ en I_0

2) $\text{ir_a}(I_0, d) = I_4$

\Rightarrow
 $\text{acción}[0, d] = \text{desp } 4$

1) $[E \rightarrow E.\text{or } T]$ en I_1

2) $\text{ir_a}(I_1, \text{or}) = I_6$

\Rightarrow
 $\text{acción}[1, \text{or}] = \text{desp } 6$

1) $[F \rightarrow .(E)]$ en I_0

2) $\text{ir_a}(I_0, E) = I_5$

\Rightarrow

$\text{acción}[0, E] = \text{desp } 5$

1) $[T \rightarrow T \text{ and } F]$ en I_2

2) $\text{ir_a}(I_2, \text{and}) = I_7$

\Rightarrow

$\text{acción}[2, \text{and}] = \text{desp } 7$

I_0	I_4	I_4
I_1	I_4	I_7
I_2		
I_3		
I_4		I_4
I_5		I_4
I_6		I_7
I_7		

$\rightarrow .id]$ en I_0 y $\text{ir_a}(0, id) = \text{desp } 1$
 $\rightarrow .(E)]$ en I_0 y $\text{ir_a}(0, (E)) = \text{desp } 2$



	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅		I ₉	I ₃	
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

- 1) E → E or T
- 2) E → T
- 3) T → T and F
- 4) T → F
- 5) F → id
- 6) F → (E)

[T → F.] en I₃

Sgte(T) = { \$, or,), and }

acción(3,\$)=red T → F=r4

acción(3,or)=red T → F=r4

acción(3,))=red T → F=r4

acción(3,and)=red T → F=r4

Tabla SLR Resultante

Regla 2.b): Si $[A \rightarrow \alpha]$ está en I_i , entonces asignar "reducir $A \rightarrow \alpha$ " a acción[i,a] para toda a en SIGUIENTE(A).
 A no puede ser S'. A a e →

Ejemplo: Diap. 46 (2 ejemplos). No se le calcula el siguiente a la gramática aumentada

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6									
2	r2	d7			r2	r2				
3										
4										
5			d4	d5						
6			d4	d5						
7			d4	d5						
8	d6				d11					
9		d7								
10										
11										

$[A \rightarrow \alpha]$ en I_i

Para cada a en $SIGTE(A)$ haga
 $acción(i, a) = \text{reducir } A \rightarrow \alpha$

$[E \rightarrow T.]$ en I_2

$SIGTE(E) = \{\$, or, ()\}$

$\Rightarrow A \neq S'$

GIC

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow id$
- 6) $F \rightarrow (E)$

	or	and	id	()	E	T	F
I_0								
I_1			I_4	I_5		I_1	I_2	I_3
I_2								
I_3			I_2					
I_4								
I_5								
I_6			I_4	I_5		I_8	I_2	I_3
I_7			I_4	I_5		I_9	I_3	
I_8			I_4	I_5		I_9		I_{10}
I_9			I_4	I_5				
I_{10}			I_6					
I_{11}			I_7			I_{11}		

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow id$
- 6) $F \rightarrow (E)$

$[E \rightarrow T.]$ en I_2 eliminado

$Sgt(E) = \{\$, or, ()\}$

$acción(2, \$) = \text{red } E \rightarrow T = r2$

$acción(2, or) = \text{red } E \rightarrow T = r2$

$acción(2, ()) = \text{red } E \rightarrow T = r2$

Tabla SLR Resultante

Se evalúan todos los estados que se logre producir mediante los siguientes pasos:

Regla 2.b): Si $[A \rightarrow \alpha]$ está en I_i , entonces asignar "reducir $A \rightarrow \alpha$ " a $acción[i, a]$ para toda a en $SIGUIENTE(A)$. A no puede ser S' .

Ejemplo: Diap 45 (2 ejemplos).

Estado	acción						ir_a
	or	and	id	()	\$	
0				d4	d5		E
1	d6						T
2			d7				F
3							
4							
5				d4	d5		
6				d4	d5		
7				d4	d5		
8	d6					d11	
9			d7				
10							
11							

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Regla 2.b): Si $[A \rightarrow \alpha.]$ está en I_i , entonces asignar “reducir $A \rightarrow \alpha$ ” a $\text{acción}[i, a]$ para toda a en $\text{SIGUIENTE}(A)$. A no puede ser S ’.

Ejemplo: Diap. 46 (2 ejemplos).

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$ *(4)*
- 5) $F \rightarrow \text{id}$
- 6) $F \rightarrow (E)$

$[T \rightarrow F.]$ en I_3

$\text{Sgte}(T) = \{\$, \text{or}, \text{), and}\}$

$\text{acción}(3, \$) = \text{red } T \rightarrow F = r4$

$\text{acción}(3, \text{or}) = \text{red } T \rightarrow F = r4$

$\text{acción}(3, \text{)}) = \text{red } T \rightarrow F = r4$

$\text{acción}(3, \text{and}) = \text{red } T \rightarrow F = r4$

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6									
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4										
5			d4	d5						
6			d4	d5						
7			d4	d5						
8	d6				d11					
9		d7								
10										
11										

$[A \rightarrow \alpha.]$ en I_1

Para cada a en $SOTE(A)$ hay,
acción(v, a) = reducir $A \rightarrow \alpha$ $\xrightarrow{\text{exc}} [A \neq S']$

GIC

$[E \rightarrow T.]$ en I_2

$SOTE(E) = \{\$, or, ()\}$

1) $E \rightarrow E$ or T

2) $E \rightarrow T$

3) $T \rightarrow T$ and F

4) $T \rightarrow F$

5) $F \rightarrow id$

6) $F \rightarrow (E)$

$[T \rightarrow F.]$ en I_3

$SOTE(T) = \{and, ;, or, ()\}$

$[F \rightarrow id.]$ en I_4

$SOTE(F) = \{and, $, or, ()\}$

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Regla 2.b): Si $[A \rightarrow \alpha.]$ está en I_i , entonces asignar “reducir $A \rightarrow \alpha$ ” a $\text{acción}[i, a]$ para toda a en $\text{SIGUIENTE}(A)$. A no puede ser S ’.

Ejemplo: Diap. 46 (2 ejemplos).

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow \text{id}$
- 6) $F \rightarrow (E)$

$[F \rightarrow \text{id}.]$ en I_4

$\text{Sgte}(F) = \{\$, \text{or},), \text{and}\}$

$\text{acción}(4, \$) = \text{red } F \rightarrow \text{id} = r5$

$\text{acción}(4, \text{or}) = \text{red } F \rightarrow \text{id} = r5$

$\text{acción}(4,)) = \text{red } F \rightarrow \text{id} = r5$

$\text{acción}(4, \text{and}) = \text{red } F \rightarrow \text{id} = r5$

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6									
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4										
5			d4	d5						
6			d4	d5						
7			d4	d5						
8	d6				d11					
9		d7								
10										
11										

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

- 1) E → E or T
- 2) E → T
- 3) T → T and F
- 4) T → F
- 5) F → id
- 6) F → (E)

[F → id.] en I₄

Sgte(F) = { \$, or,), and }

acción(4,\$)=red F → id=r5

acción(4,or)= red F → id=r5

acción(4,))= red F → id=r5

acción(4, and)= red F → id=r5

Tabla SLR Resultante

Regla 2.b): Si [A → α.] está en I_i, entonces asignar “reducir A → α” a acción[i,a] para toda a en SIGUIENTE(A). A no puede ser S’.

Ejemplo: Diap. 46 (2 ejemplos).

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6									
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5						
6			d4	d5						
7			d4	d5						
8	d6				d11					
9		d7								
10										
11										

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Regla 2.b): Si $[A \rightarrow \alpha.]$ está en I_i , entonces asignar “reducir $A \rightarrow \alpha$ ” a $\text{acción}[i, a]$ para toda a en $\text{SIGUIENTE}(A)$. A no puede ser S ’.

Ejemplo: [Diap. 51](#) (1 ejemplo)

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow \text{id}$
- 6) $F \rightarrow (E)$

$[E \rightarrow E \text{ or } T.]$ en I_9

$Sgte(E) = \{\$, \text{or},)\}$

$\text{acción}(9, \$) = \text{red } E \rightarrow E \text{ or } T = r1$

$\text{acción}(9, \text{or}) = \text{red } E \rightarrow E \text{ or } T = r1$

$\text{acción}(9,)) = \text{red } E \rightarrow E \text{ or } T = r1$

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6									
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5						
6			d4	d5						
7			d4	d5						
8	d6				d11					
9		d7								
10										
11										

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Regla 2.b): Si $[A \rightarrow \alpha.]$ está en I_i , entonces asignar “reducir $A \rightarrow \alpha$ ” a $\text{acción}[i, a]$ para toda a en $\text{SIGUIENTE}(A)$. A no puede ser S ’.

Ejemplo: Diap. 51 (1 ejemplo)

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow \text{id}$
- 6) $F \rightarrow (E)$

$[E \rightarrow E \text{ or } T.]$ en I_9

$\text{Sgte}(E) = \{\$, \text{or},)\}$

$\text{acción}(9, \$) = \text{red } E \rightarrow E \text{ or } T = r1$

$\text{acción}(9, \text{or}) = \text{red } E \rightarrow E \text{ or } T = r1$

$\text{acción}(9,)) = \text{red } E \rightarrow E \text{ or } T = r1$

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6									
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5						
6			d4	d5						
7			d4	d5						
8	d6				d11					
9	r1	d7			r1	r1				
10										
11										

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

- 1) E → E or T
- 2) E → T
- 3) T → T and F
- 4) T → F
- 5) F → id
- 6) F → (E)

[T → T and F] en I₁₀

Sgte(T) = { \$, or,), and }

acción(10,\$)=red T → T and F=r3

acción(10,or)=red T → T and F=r3

acción(10,))=red T → T and F=r3

acción(10,and)= red T → T and F=r3

Tabla SLR Resultante

Regla 2.b): Si [A → α.] está en I_i, entonces asignar “reducir A → α” a acción[i,a] para toda a en SIGUIENTE(A). A no puede ser S’.

Ejemplo: [Diap. 53](#) (1 ejemplo).

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6									
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5						
6			d4	d5						
7			d4	d5						
8	d6				d11					
9	r1	d7			r1	r1				
10										
11										

Tabla SLR Resultante

Regla 2.b): Si $[A \rightarrow \alpha.]$ está en I_i , entonces asignar “reducir $A \rightarrow \alpha$ ” a $acción[i, a]$ para toda a en $SIGUIENTE(A)$. A no puede ser S ’.

Ejemplo: Diap. 53 (1 ejemplo).

- 1) $E \rightarrow E$ or T
 - 2) $E \rightarrow T$
 - 3) $T \rightarrow T$ and F
 - 4) $T \rightarrow F$
 - 5) $F \rightarrow id$
 - 6) $F \rightarrow (E)$

[T → TandF.] en I₁₀

Sgte(T) = { \$, or,), and }

acción(10,\$)=red T →TandF=r3

acción(10,or)=red T →TandF=r3

acción(10.)=red T →TandF=r3

Tabla SLR Resultante

Regla 2.b): Si $[A \rightarrow \alpha.]$ está en I_i , entonces asignar “reducir $A \rightarrow \alpha$ ” a $acción[i, a]$ para toda a en $SIGUIENTE(A)$. A no puede ser S ’.

Ejemplo: Diap. 54 (1 ejemplo).

- 1) $E \rightarrow E \text{ or } T$
 - 2) $E \rightarrow T$
 - 3) $T \rightarrow T \text{ and } F$
 - 4) $T \rightarrow F$
 - 5) $F \rightarrow \text{id}$
 - 6) $F \rightarrow (E)$

[F → (E).] en **I₁₁**

Sgte(F) = { \$, or,), and }

acción(11,\$)=red F →(E)=r6

acción(11,or)= red F →(E)=r6

acción(11.))= red F →(E)=r6

acción(11.and)= red F →(E)=r6

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Regla 2.b): Si $[A \rightarrow \alpha.]$ está en I_i , entonces asignar “reducir $A \rightarrow \alpha$ ” a $\text{acción}[i, a]$ para toda a en $\text{SIGUIENTE}(A)$. A no puede ser S ’.

Ejemplo: Diap. 54 (1 ejemplo).

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow \text{id}$
- 6) $F \rightarrow (E)$

$[F \rightarrow (E).]$ en I_{11}

Sgte(F) = { $$$, or, and}

acción(11,\$)=red $F \rightarrow (E) = r6$

acción(11,or)= red $F \rightarrow (E) = r6$

acción(11,))= red $F \rightarrow (E) = r6$

acción(11, and)= red $F \rightarrow (E) = r6$

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6									
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5						
6			d4	d5						
7			d4	d5						
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Regla 2.c): Si $[S' \rightarrow S.]$ está en I_i , entonces asignar “aceptar” a $acción[i, \$]$.

Ejemplo: Diap. 45 (ejemplo).

- 1) E → E or T
- 2) E → T
- 3) T → T and F
- 4) T → F
- 5) F → id
- 6) F → (E)

$[E' \rightarrow E.]$ en I₁

acción(1,\$)=aceptar

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6									
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5						
6			d4	d5						
7			d4	d5						
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				80

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅			I ₉	I ₃
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Regla 2.c): Si $[S' \rightarrow S.]$ está en I_i , entonces asignar “aceptar” a $\text{acción}[i, \$]$.

Ejemplo: Diap. 45 (ejemplo).

- 1) E → E or T
- 2) E → T
- 3) T → T and F
- 4) T → F
- 5) F → id
- 6) F → (E)

$[E' \rightarrow E.]$ en I₁

acción(1,\$)=aceptar

Vna lñen
por cada produccn

Estado	acción						ir_a		
	or	and	id	()	\$	E	T	F
0			d4	d5					
1	d6								
2	r2	d7				r2	r2		
3	r4	r4				r4	r4		
4	r5	r5				r5	r5		
5			d4	d5					
6			d4	d5					
7			d4	d5					
8	d6					d11			
9	r1	d7				r1	r1		
10	r3	r3				r3	r3		
11	r6	r6				r6	r6		

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅		I ₄	I ₅		I ₈	I ₂	I ₃	
I ₆		I ₄	I ₅			I ₉	I ₃	
I ₇		I ₄	I ₅				I ₁₀	
I ₈	I ₆			I ₁₁				
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

escribir los números
de los I para los term. nulos

Regla 3): Las transiciones ir_a para el estado i se construyen para todos los terminales A así: Si $Ir_a(I_i, A) = I_j$, entonces $ir_a[i, A] = j$.

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5						
1	d6					Acept				
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5						
6			d4	d5						
7			d4	d5						
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				82

	or	and	id	()	E	T	F
I ₀			I ₄	I ₅		I ₁	I ₂	I ₃
I ₁	I ₆							
I ₂		I ₇						
I ₃								
I ₄								
I ₅			I ₄	I ₅		I ₈	I ₂	I ₃
I ₆			I ₄	I ₅		I ₉	I ₃	
I ₇			I ₄	I ₅				I ₁₀
I ₈	I ₆				I ₁₁			
I ₉		I ₇						
I ₁₀								
I ₁₁								

Tabla SLR Resultante

Regla 3): Las transiciones ir_a para el estado i se construyen para todos los terminales A así: Si $Ir_a(I_i, A) = I_j$, entonces $ir_a[i, A] = j$.

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5			1	2	3	
1	d6					Acept				
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5			8	2	3	
6			d4	d5				9	3	
7			d4	d5					10	
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				83

Modelo de Análisis Sintáctico LR

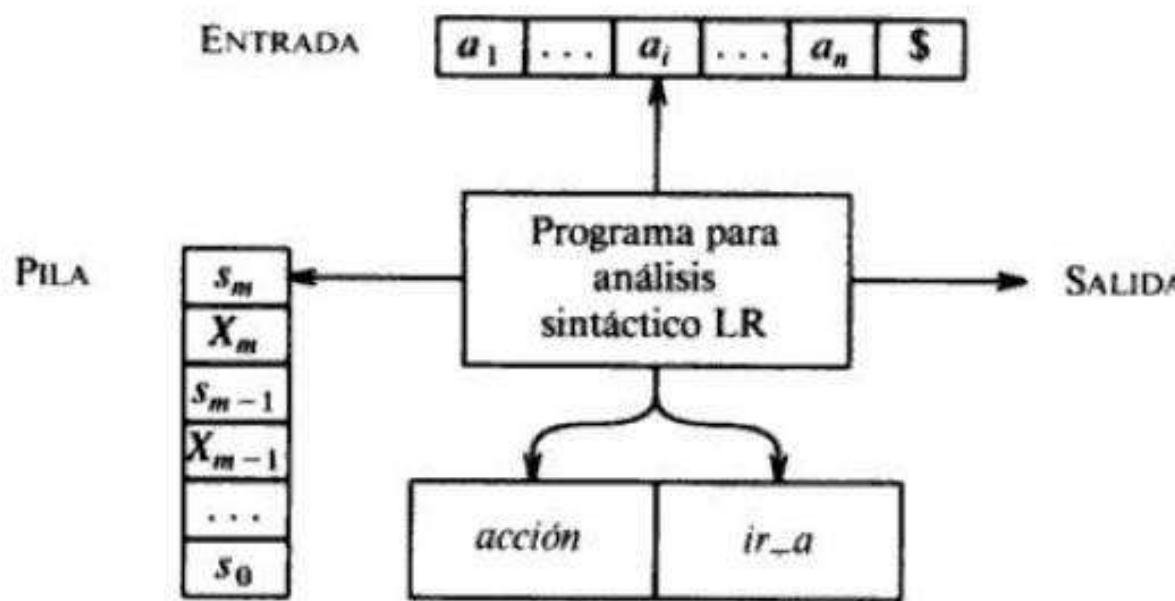


Fig. 4.29. Modelo de un analizador sintáctico LR.

Algoritmo de Análisis Sintáctico

```
apuntar ae al primer símbolo de w$;  
repeat forever begin  
    sea s el estado en la cima de la pila y  
        a el símbolo apuntado por ae;  
    if acción [s, a] = desplazar s' then begin  
        meter a y después s' en la cima de la pila;  
        avanzar ae al siguiente símbolo de entrada  
    end  
    else if acción [s, a] = reducir A → β then begin  
        sacar 2* | β | símbolos de la pila;  
        sea s' el estado que ahora está en la cima de la pila;  
        meter A y después ir_a [s', A] en la cima de la pila;  
        emitir la producción A → β  
    end  
    else if acción [s, a] = aceptar then  
        return  
    else error ()  
end
```

Fig. 4.30. Programa para análisis sintáctico LR

Algoritmo SLR

- 1) $E \rightarrow E$ or T
 - 2) $E \rightarrow T$
 - 3) $T \rightarrow T$ and R
 - 4) $T \rightarrow F$
 - 5) $F \rightarrow id$
 - 6) $F \rightarrow (E)$

apuntar *ae* al primer simbolo de *w\$*:

repeat forever begin

sea s el estado en la cima de la pila y

a el símbolo apuntado por ae;

if *acción* [*s, a*] = desplazar *s'* **then begin**

meter a y después s' en la cima de la pila;

avanzar *ae* al siguiente símbolo de entrada

end

else if *acción* [*s, a*] = reducir $A \rightarrow \beta$ **then begin**

sacar $2 * |\beta|$ símbolos de la pila;

sea s' el estado que ahora está en la cima de la pila;

meter A y después $ir_a [s', A]$ en la cima de la pila;

emitir la producción $A \rightarrow \beta$

end

else if *acción* [*s, a*] = aceptar **then**

return

else error()

	acción							ir_a		
Estado	or	and	id	()	\$	E	T	F	
0			d4	d5			1	2	3	
1	d6					Acept				
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5			8	2	3	
6			d4	d5				9	3	
7			d4	d5					10	
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				

Algoritmo SLR

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow \text{id}$
- 6) $F \rightarrow (E)$

apuntar ae al primer símbolo de $w\$$;

repeat forever begin

 sea s el estado en la cima de la pila y

a el símbolo apuntado por ae ;

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

 meter a y después s' en la cima de la pila;

 avanzar ae al siguiente símbolo de entrada

end

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

 sacar $2 * |\beta|$ símbolos de la pila;

 sea s' el estado que ahora está en la cima de la pila;

 meter A y después $ir_a [s', A]$ en la cima de la pila;

 emitir la producción $A \rightarrow \beta$

end

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

return

else error ()

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5			1	2	3	
1	d6					Acept				
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5			8	2	3	
6			d4	d5				9	3	
7			d4	d5					10	
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				

Pila	Entrada	Acción
0	id or id and id\$	desplazar 4
0id4	or id and id\$	red. F→id

Algoritmo SLR

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow \text{id}$
- 6) $F \rightarrow (E)$

apuntar ae al primer símbolo de $w\$$;

repeat forever begin

 sea s el estado en la cima de la pila y

a el símbolo apuntado por ae ;

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

 meter a y después s' en la cima de la pila;

 avanzar ae al siguiente símbolo de entrada

end

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

 sacar $2 * |\beta|$ símbolos de la pila;

 sea s' el estado que ahora está en la cima de la pila;

 meter A y después $ir_a [s', A]$ en la cima de la pila;

 emitir la producción $A \rightarrow \beta$

end

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

return

else error ()

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5			1	2	3	
1	d6					Acept				
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5			8	2	3	
6			d4	d5				9	3	
7			d4	d5					10	
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				

Pila	Entrada	Acción
0	id or id and id\$	desplazar 4
0id4	or id and id\$	red. F→id
0F3	or id and id\$	red. T→F

Algoritmo SLR

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow \text{id}$
- 6) $F \rightarrow (E)$

apuntar ae al primer símbolo de $w\$$;

repeat forever begin

 sea s el estado en la cima de la pila y

a el símbolo apuntado por ae ;

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

 meter a y después s' en la cima de la pila;

 avanzar ae al siguiente símbolo de entrada

end

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

 sacar $2 * |\beta|$ símbolos de la pila;

 sea s' el estado que ahora está en la cima de la pila;

 meter A y después $ir_a [s', A]$ en la cima de la pila;

 emitir la producción $A \rightarrow \beta$

end

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

return

else error ()

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5			1	2	3	
1	d6					Acept				
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5			8	2	3	
6			d4	d5				9	3	
7			d4	d5					10	
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				

Pila	Entrada	Acción
0	id or id and id\$	desplazar 4
0id4	or id and id\$	red. F→id
0F3	or id and id\$	red. T→F
0T2	or id and id\$	red. E→T

Algoritmo SLR

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow id$
- 6) $F \rightarrow (E)$

apuntar ae al primer símbolo de $w\$$;

repeat forever begin

 sea s el estado en la cima de la pila y

a el símbolo apuntado por ae ;

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

 meter a y después s' en la cima de la pila;

 avanzar ae al siguiente símbolo de entrada

end

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

 sacar $2 * |\beta|$ símbolos de la pila;

 sea s' el estado que ahora está en la cima de la pila;

 meter A y después $ir_a [s', A]$ en la cima de la pila;

 emitir la producción $A \rightarrow \beta$

end

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

return

else error ()

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5			1	2	3	
1	d6					Acept				
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5			8	2	3	
6			d4	d5				9	3	
7			d4	d5					10	
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				

Pila	Entrada	Acción
0	id or id and id\$	desplazar 4
0id4	or id and id\$	red. F→id
0F3	or id and id\$	red. T→F
0T2	or id and id\$	red. E→T
0E1	or id and id\$	desplazar 6

Algoritmo SLR

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow \text{id}$
- 6) $F \rightarrow (E)$

apuntar ae al primer símbolo de $w\$$;

repeat forever begin

 sea s el estado en la cima de la pila y

a el símbolo apuntado por ae ;

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

 meter a y después s' en la cima de la pila;

 avanzar ae al siguiente símbolo de entrada

end

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

 sacar $2 * |\beta|$ símbolos de la pila;

 sea s' el estado que ahora está en la cima de la pila;

 meter A y después $ir_a [s', A]$ en la cima de la pila;

 emitir la producción $A \rightarrow \beta$

end

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

return

else error ()

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5			1	2	3	
1	d6						Acept			
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5			8	2	3	
6			d4	d5				9	3	
7			d4	d5					10	
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				

Pila	Entrada	Acción
0	id or id and id\$	desplazar 4
0id4	or id and id\$	red. F→id
0F3	or id and id\$	red. T→F
0T2	or id and id\$	red. E→T
0E1	or id and id\$	desplazar 6
0E1or6	id and id\$	desplazar 4

Algoritmo SLR

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow \text{id}$
- 6) $F \rightarrow (E)$

apuntar ae al primer símbolo de $w\$$;

repeat forever begin

 sea s el estado en la cima de la pila y

a el símbolo apuntado por ae ;

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

 meter a y después s' en la cima de la pila;

 avanzar ae al siguiente símbolo de entrada

end

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

 sacar $2 * |\beta|$ símbolos de la pila;

 sea s' el estado que ahora está en la cima de la pila;

 meter A y después $ir_a [s', A]$ en la cima de la pila;

 emitir la producción $A \rightarrow \beta$

end

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

return

else error ()

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5			1	2	3	
1	d6						Acept			
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5			8	2	3	
6			d4	d5				9	3	
7			d4	d5					10	
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				

Pila	Entrada	Acción
0	id or id and id\$	desplazar 4
0id4	or id and id\$	red. F→id
0F3	or id and id\$	red. T→F
0T2	or id and id\$	red. E→T
0E1	or id and id\$	desplazar 6
0E1or6	id and id\$	desplazar 4
0E1or6id4	and id\$	red. F→id

Algoritmo SLR

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow id$
- 6) $F \rightarrow (E)$

apuntar ae al primer símbolo de $w\$$;

repeat forever begin

 sea s el estado en la cima de la pila y

a el símbolo apuntado por ae ;

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

 meter a y después s' en la cima de la pila;

 avanzar ae al siguiente símbolo de entrada

end

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

 sacar $2 * |\beta|$ símbolos de la pila;

 sea s' el estado que ahora está en la cima de la pila;

 meter A y después $ir_a [s', A]$ en la cima de la pila;

 emitir la producción $A \rightarrow \beta$

end

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

return

else error ()

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5			1	2	3	
1	d6						Acept			
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5			8	2	3	
6			d4	d5				9	3	
7			d4	d5					10	
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				

Pila	Entrada	Acción
0	id or id and id\$	desplazar 4
0id4	or id and id\$	red. F→id
0F3	or id and id\$	red. T→F
0T2	or id and id\$	red. E→T
0E1	or id and id\$	desplazar 6
0E1or6	id and id\$	desplazar 4
0E1or6id4	and id\$	red. F→id
0E1or6F3	and id\$	red. T→F

Algoritmo SLR

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow id$
- 6) $F \rightarrow (E)$

apuntar ae al primer símbolo de $w\$$;

repeat forever begin

 sea s el estado en la cima de la pila y

a el símbolo apuntado por ae ;

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

 meter a y después s' en la cima de la pila;

 avanzar ae al siguiente símbolo de entrada

end

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

 sacar $2 * |\beta|$ símbolos de la pila;

 sea s' el estado que ahora está en la cima de la pila;

 meter A y después $ir_a [s', A]$ en la cima de la pila;

 emitir la producción $A \rightarrow \beta$

end

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

return

else error ()

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5			1	2	3	
1	d6					Acept				
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5			8	2	3	
6			d4	d5				9	3	
7			d4	d5					10	
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				

Pila	Entrada	Acción
0id4	or id and id\$	red. F→id
0F3	or id and id\$	red. T→F
0T2	or id and id\$	red. E→T
0E1	or id and id\$	desplazar 6
0E1or6	id and id\$	desplazar 4
0E1or6id4	and id\$	red. F→id
0E1or6F3	and id\$	red. T→F
0E1or6T9	and id\$	desplazar 7

Algoritmo SLR

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow id$
- 6) $F \rightarrow (E)$

apuntar ae al primer símbolo de $w\$$;

repeat forever begin

 sea s el estado en la cima de la pila y

a el símbolo apuntado por ae ;

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

 meter a y después s' en la cima de la pila;

 avanzar ae al siguiente símbolo de entrada

end

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

 sacar $2 * |\beta|$ símbolos de la pila;

 sea s' el estado que ahora está en la cima de la pila;

 meter A y después $ir_a [s', A]$ en la cima de la pila;

 emitir la producción $A \rightarrow \beta$

end

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

return

else error ()

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5			1	2	3	
1	d6					Acept				
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5			8	2	3	
6			d4	d5				9	3	
7			d4	d5					10	
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				

Pila	Entrada	Acción
0F3	or id and id\$	red. T→F
0T2	or id and id\$	red. E→T
0E1	or id and id\$	desplazar 6
0E1or6	id and id\$	desplazar 4
0E1or6id4	and id\$	red. F→id
0E1or6F3	and id\$	red. T→F
0E1or6T9	and id\$	desplazar 7
0E1or6T9and7	id\$	desplazar 4

Algoritmo SLR

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow id$
- 6) $F \rightarrow (E)$

apuntar ae al primer símbolo de $w\$$;

repeat forever begin

 sea s el estado en la cima de la pila y

a el símbolo apuntado por ae ;

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

 meter a y después s' en la cima de la pila;

 avanzar ae al siguiente símbolo de entrada

end

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

 sacar $2 * |\beta|$ símbolos de la pila;

 sea s' el estado que ahora está en la cima de la pila;

 meter A y después $ir_a [s', A]$ en la cima de la pila;

 emitir la producción $A \rightarrow \beta$

end

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

return

else error ()

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5			1	2	3	
1	d6					Acept				
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5			8	2	3	
6			d4	d5				9	3	
7			d4	d5					10	
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				

Pila	Entrada	Acción
0T2	or id and id\$	red. $E \rightarrow T$
0E1	or id and id\$	desplazar 6
0E1or6	id and id\$	desplazar 4
0E1or6id4	and id\$	red. $F \rightarrow id$
0E1or6F3	and id\$	red. $T \rightarrow F$
0E1or6T9	and id\$	desplazar 7
0E1or6T9and7	id\$	desplazar 4
0E1or6T9and7id4	\$	red. $F \rightarrow id$

Algoritmo SLR

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow id$
- 6) $F \rightarrow (E)$

apuntar ae al primer símbolo de $w\$$;

repeat forever begin

 sea s el estado en la cima de la pila y

a el símbolo apuntado por ae ;

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

 meter a y después s' en la cima de la pila;

 avanzar ae al siguiente símbolo de entrada

end

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

 sacar $2 * |\beta|$ símbolos de la pila;

 sea s' el estado que ahora está en la cima de la pila;

 meter A y después $ir_a [s', A]$ en la cima de la pila;

 emitir la producción $A \rightarrow \beta$

end

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

return

else error ()

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5			1	2	3	
1	d6					Acept				
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5			8	2	3	
6			d4	d5				9	3	
7			d4	d5					10	
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				

Pila	Entrada	Acción
0E1	or id and id\$	desplazar 6
0E1or6	id and id\$	desplazar 4
0E1or6id4	and id\$	red. F→id
0E1or6F3	and id\$	red. T→F
0E1or6T9	and id\$	desplazar 7
0E1or6T9and7	id\$	desplazar 4
0E1or6T9and7id4	\$	red. F→id
0E1or6T9and7F10	\$	red. T→T and F

Algoritmo SLR

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow id$
- 6) $F \rightarrow (E)$

apuntar ae al primer símbolo de $w\$$;
repeat forever begin

 sea s el estado en la cima de la pila y
 a el símbolo apuntado por ae ;

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

 meter a y después s' en la cima de la pila;

 avanzar ae al siguiente símbolo de entrada

end

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

 sacar $2 * |\beta|$ símbolos de la pila;

 sea s' el estado que ahora está en la cima de la pila;

 meter A y después $ir_a [s', A]$ en la cima de la pila;

 emitir la producción $A \rightarrow \beta$

end

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

return

else error ()

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5			1	2	3	
1	d6					Acept				
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5			8	2	3	
6			d4	d5				9	3	
7			d4	d5					10	
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				

Pila	Entrada	Acción
0E1or6	id and id\$	desplazar 4
0E1or6id4	and id\$	red. F→id
0E1or6F3	and id\$	red. T→F
0E1or6T9	and id\$	desplazar 7
0E1or6T9and7	id\$	desplazar 4
0E1or6T9and7id4	\$	red. F→id
0E1or6T9and7F10	\$	red. T→T and F
0E1or6T9	\$	red. E→E or T

Algoritmo SLR

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow id$
- 6) $F \rightarrow (E)$

apuntar ae al primer símbolo de $w\$$;

repeat forever begin

 sea s el estado en la cima de la pila y

a el símbolo apuntado por ae ;

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

 meter a y después s' en la cima de la pila;

 avanzar ae al siguiente símbolo de entrada

end

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

 sacar $2 * |\beta|$ símbolos de la pila;

 sea s' el estado que ahora está en la cima de la pila;

 meter A y después $ir_a [s', A]$ en la cima de la pila;

 emitir la producción $A \rightarrow \beta$

end

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

return

else error ()

Estado	acción							ir_a		
	or	and	id	()	\$	E	T	F	
0			d4	d5			1	2	3	
1	d6					Acept				
2	r2	d7			r2	r2				
3	r4	r4			r4	r4				
4	r5	r5			r5	r5				
5			d4	d5			8	2	3	
6			d4	d5				9	3	
7			d4	d5					10	
8	d6				d11					
9	r1	d7			r1	r1				
10	r3	r3			r3	r3				
11	r6	r6			r6	r6				

Pila	Entrada	Acción
0E1or6id4	and id\$	red. $F \rightarrow id$
0E1or6F3	and id\$	red. $T \rightarrow F$
0E1or6T9	and id\$	desplazar 7
0E1or6T9and7	id\$	desplazar 4
0E1or6T9and7id4	\$	red. $F \rightarrow id$
0E1or6T9and7F10	\$	red. $T \rightarrow T \text{ and } F$
0E1or6T9	\$	red. $E \rightarrow E \text{ or } T$
0E1	\$	aceptar

Algoritmo SLR

- 1) $E \rightarrow E \text{ or } T$
- 2) $E \rightarrow T$
- 3) $T \rightarrow T \text{ and } F$
- 4) $T \rightarrow F$
- 5) $F \rightarrow id$
- 6) $F \rightarrow (E)$

¿Cuál sería el Árbol de Análisis Sintáctico Ascendente?



Posic	Pila	Entrada	Acción
1	0	id or id and id\$	desplazar 4
2	0id4	or id and id\$	red. $F \rightarrow id$
3	0F3	or id and id\$	red. $T \rightarrow F$
4	0T2	or id and id\$	red. $E \rightarrow T$
5	0E1	or id and id\$	desplazar 6
6	0E1or6	id and id\$	desplazar 4
7	0E1or6id4	and id\$	red. $F \rightarrow id$
8	0E1or6F3	and id\$	red. $T \rightarrow F$
9	0E1or6T9	and id\$	desplazar 7
10	0E1or6T9and7	id\$	desplazar 4
11	0E1or6T9and7id4	\$	red. $F \rightarrow id$
12	0E1or6T9and7F10	\$	red. $T \rightarrow T \text{ and } F$
13	0E1or6T9	\$	red. $E \rightarrow E \text{ or } T$
14	0E1	\$	aceptar

14 13 12 9,10,11 8 5,6,7 4
 $E \Rightarrow E \text{ or } T \Rightarrow E \text{ or } T \text{ and } F \Rightarrow E \text{ or } T \text{ and } id \Rightarrow E \text{ or } F \text{ and } id \Rightarrow E \text{ or } id \text{ and } id \Rightarrow T \text{ or } id \text{ and } id \Rightarrow$
 3 1,2
 $F \text{ or } id \text{ and } id \Rightarrow id \text{ or } id \text{ and } id$

$[A \rightarrow \alpha.]$ en I_1

Para cada a en $SOTE(A)$ haga
acción(i, a) = reducir $A \rightarrow \alpha$

exc $\boxed{A \neq S'}$

GIC

$[E \rightarrow T.]$ en I_2

$SOTE(E) = \{\$, or, ()\}$

VE $\rightarrow E_{or} T$

2) $E \rightarrow T$

3) $T \rightarrow T_{and} F$

4) $T \rightarrow F$

5) $F \rightarrow id$

6) $F \rightarrow (E)$

$[T \rightarrow F.]$ en I_3

$SOTE(T) = \{and, \$, or, ()\}$

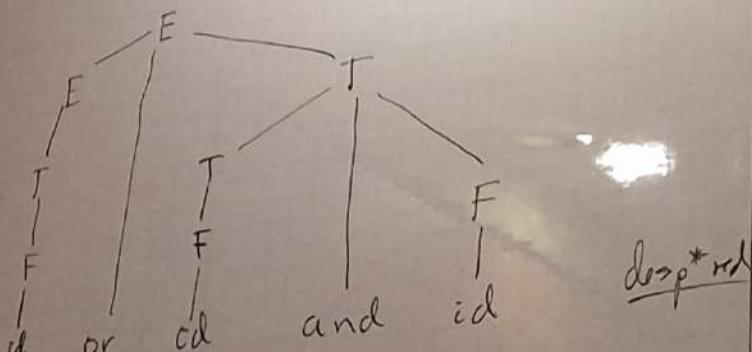
$[F \rightarrow id.]$ en I_4

$SOTE(F) = \{and, \$, or, ()\}$

$[E \rightarrow Eur]$ en I_5

LALR

LR-Canónico



$E \Rightarrow Eur \rightarrow T \Rightarrow Eur \rightarrow T \rightarrow F \Rightarrow Eur \rightarrow T \rightarrow and \rightarrow F \Rightarrow Eur \rightarrow T \rightarrow and \rightarrow id \Rightarrow id$

$E \rightarrow F \rightarrow id \Rightarrow Eur \rightarrow id \rightarrow T \rightarrow id \rightarrow and \rightarrow F \rightarrow id \rightarrow id \Rightarrow id \rightarrow id \rightarrow id \rightarrow id$