

Autómatas y Lenguajes

3^{er} curso
1^{er} cuatrimestre

Alfonso Ortega: alfonso.ortega@uam.es

UNIDAD 2: Procesadores de lenguaje

TEMA Análisis sintáctico

- a) Análisis sintáctico ascendente SLR(1): primer paso LR(0)
- b) Análisis sintáctico ascendente SLR(1): modificación a tablas (LR0)

Tema b: Analizador sintáctico SLR(1)

- b.1 Introducción
- b.2 Conceptos previos
- b.3 Construcción de la tabla de análisis
- b.4 Uso de la tabla
- b.5 Limitaciones de la técnica
- b.6 Ejercicios de repaso



b.1

Introducción

Análisis LR(k)

Introducción (I)

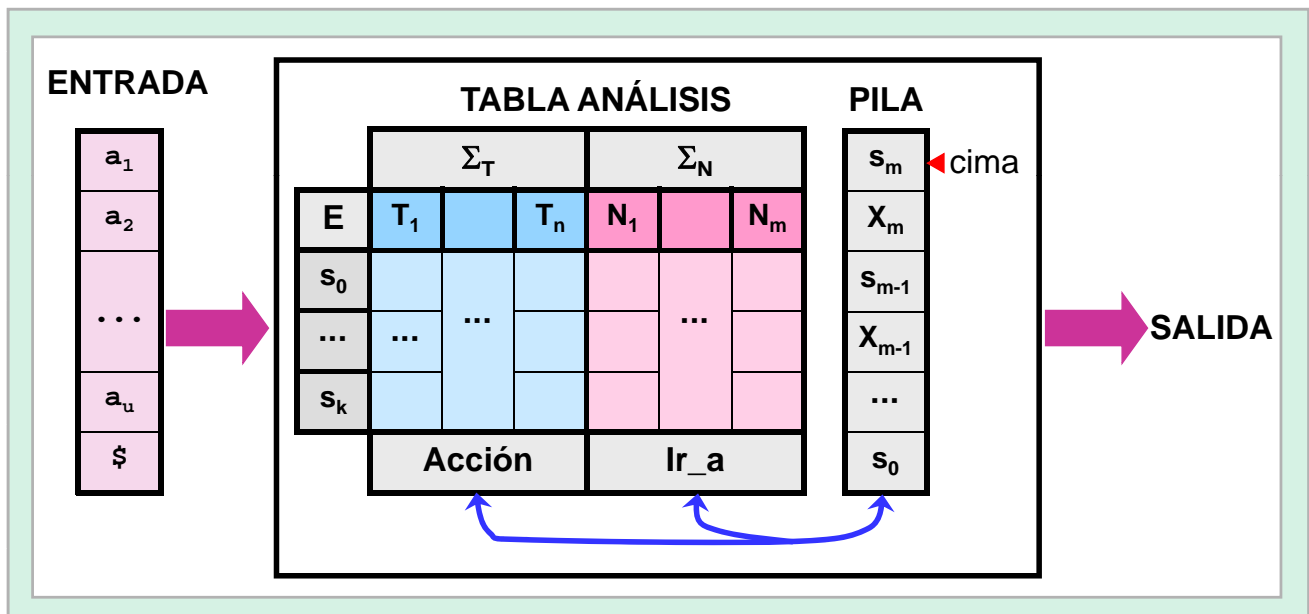
- Recordemos que un analizador LR(k) es aquel que:
 - Es capaz de analizar una gramática LR(k), esto es independiente del contexto no ambigua.
 - Realiza un análisis sintáctico ascendente (bottom-up).
 - Procesa la entrada de izquierda a derecha (Left-to-Right)
 - Aplica derivaciones más a la derecha (Rightmost).
 - Considera k símbolos antes de realizar una acción.
- La entrada como la de cualquier analizador sintáctico es la cadena de tokens resultado del análisis morfológico.
- La salida puede ser:
 - Aceptar la cadena si cumple la gramática.
 - Rechazarla en caso de que no la cumpla (esto es, la cadena no se reduce al axioma).

Análisis LR(k)

Introducción (II)

- En el tema pasado se estudiaron los analizadores LR(0).
- Estos analizadores son útiles para lenguajes que se pueden expresar con gramáticas LR(0), en las cuales siempre se puede decidir entre realizar la operación de desplazar o reducir.
- Pero, por ejemplo con la gramática:

```
(1) Stmt :- if Expr then Stmt .
(2) Stmt :- if Expr then Stmt . else Stmt
```
- Ya no se podría usar un analizador LR(0) porque en este estado, no sabría si reducir usando (1) o bien desplazar a (2).
- Siempre que esto ocurre, es necesario realizar un análisis que tenga en cuenta al menos el símbolo siguiente.
- Esto es, un análisis LR(1). En este tema vamos a estudiar la versión más simple de analizador LR(1): Simple LR(1) o SLR(1).



b.2

Conceptos previos

Formalización: conjunto primero

Definición

- Formalmente:

Sea $G = \{\Sigma_N, \Sigma_T, P, A\}$ una gramática independiente del contexto
Si α es una forma sentencial de la gramática ($\alpha \in (\Sigma_N \cup \Sigma_T)^*$) llamaremos $\text{primeros}(\alpha)$ al “conjunto de terminales que inician las cadenas derivadas de α ” y, en el caso de que desde α se pueda derivar la cadena vacía (λ), entonces ésta (λ) también pertenecerá a $\text{primeros}(\alpha)$

Formalización: conjunto primero

Algoritmo de cálculo

- Describiremos ahora el algoritmo de cálculo de $\text{primeros}(X) \quad \forall X \in \Sigma_N \cup \Sigma_T$:

Se tiene que aplicar las siguientes reglas hasta que no se puedan añadir más símbolos terminales o λ

1. Si $X \in \Sigma_T$ entonces $\text{primeros}(X) = \{X\}$.
2. Si $X \rightarrow \lambda \in P$ entonces $\lambda \in \text{primeros}(X)$.
3. Si $X \rightarrow Y_1 Y_2 \dots Y_k \in P$ entonces
 - Si $\lambda \in \text{primeros}(Y_j) \quad \forall j \in \{1, 2, \dots, k\}$ entonces $\lambda \in \text{primeros}(X)$.
 - Sea j el primer subíndice que cumple $\lambda \notin \text{primeros}(Y_j)$ entonces $\text{primeros}(Y_i) - \{\lambda\} \subseteq \text{primeros}(X) \quad \forall i \in \{1, 2, \dots, j\}$ obsérvese que para $i=j$ se cumple que $\text{primeros}(Y_j) - \{\lambda\} = \text{primeros}(Y_j)$ ya que $\lambda \notin \text{primeros}(Y_j)$

Formalización: conjunto primero

Algoritmo de cálculo

- Describiremos ahora el algoritmo de cálculo de $\text{primero}(\alpha) \quad \forall \alpha \in (\Sigma_N \cup \Sigma_T)^*$:

Sea $\alpha = X_1 X_2 \dots X_n$ entonces

- Si $\lambda \in \text{primero}(X_i) \quad \forall i \in \{1, 2, \dots, n\}$ entonces $\lambda \in \text{primero}(\alpha)$.
- Sea j el primer subíndice que cumple $\lambda \notin \text{primero}(X_j)$ entonces $\text{primero}(X_i) - \{\lambda\} \subseteq \text{primero}(\alpha) \quad \forall i \in \{1, 2, \dots, j\}$ obsérvese que para $i=j$ se cumple que $\text{primero}(X_j) - \{\lambda\} = \text{primero}(X_j)$ ya que $\lambda \notin \text{primero}(X_j)$

Conjunto primero

Algoritmo de cálculo

- Informalmente, cómo calcular $\text{primero}(X)$:
- Para todas la reglas donde el símbolo no terminal X aparezca en la parte izquierda (las reglas serán de la forma $X \rightarrow S_1 S_2 \dots S_n$), se aplica uno de los tres casos siguientes:
 1. Si S_1 es un símbolo terminal entonces S_1 está en $\text{primero}(X)$
 2. Si la regla es de la forma $X \rightarrow \lambda$ entonces λ está en $\text{primero}(X)$
 3. Se comprueban todas las reglas donde S_1 esté en la parte izquierda (sin aplicar el caso 2). Si $\text{primero}(S_1)$ contiene λ entonces también se comprueban todas las reglas donde S_2 esté en la parte izquierda. Si ocurriera que $\text{primero}(S_2)$ contiene λ se procedería con S_3 y así sucesivamente. Si se llegara a S_n , entonces λ también está en $\text{primero}(X)$

Formalización: conjunto siguiente

Definición

- Formalmente:

Sea $G = \{\Sigma_N, \Sigma_T, P, A\}$ una gramática independiente del contexto
Si x es un símbolo no terminal de la gramática ($x \in \Sigma_N$) llamaremos $\text{siguiente}(x)$ al “conjunto de terminales que pueden aparecer inmediatamente a la derecha de x en alguna forma sentencial”. Si x puede ser el símbolo situado más a la derecha de una forma sentencial, entonces $\$ \in \text{siguiente}(x)$

13

Formalización: conjunto siguiente

Algoritmo de cálculo

- Describiremos ahora el algoritmo de cálculo de $\text{siguiente}(x) \quad \forall x \in \Sigma_N$:

Se aplica los siguientes pasos

1. Si $x=A$ (axioma) $\$ \in \text{siguiente}(x)$.
2. $\forall Y \rightarrow \alpha x \beta \in P, \beta \neq \lambda \wedge \lambda \notin \text{primero}(\beta) \Rightarrow \text{primero}(\beta) \subseteq \text{siguiente}(x)$.
3. $\forall Y \rightarrow \alpha x \beta \in P, \beta \neq \lambda \wedge \lambda \in \text{primero}(\beta)$.
 - Entonces $\text{primero}(\beta) - \{\lambda\} \subseteq \text{siguiente}(x)$.
 - $Y \text{ siguiente}(Y) \subseteq \text{siguiente}(x)$.
4. $\forall Y \rightarrow \alpha x \in P \Rightarrow \text{siguiente}(Y) \subseteq \text{siguiente}(x)$.

14

Conjunto siguiente

Algoritmo de cálculo

- Informalmente, cómo calcular $\text{siguiente}(X)$:
- Para todas la reglas donde el símbolo no terminal X aparezca en la parte derecha (las reglas serán de la forma $Y \rightarrow S_1 \dots S_i X S_{j+1} \dots S_n$), se aplica uno de los cuatro casos siguientes:
 1. Si X es el axioma, entonces el símbolo $\$$ está en $\text{siguiente}(X)$
 2. Si $\text{primero}(S_j)$ NO contiene a λ , entonces $\text{primero}(S_j)$ está en $\text{siguiente}(X)$
 3. Si $\text{primero}(S_j)$ SÍ contiene a λ , entonces $\text{primero}(S_j)$ sin λ y $\text{siguiente}(Y)$ están en $\text{siguiente}(X)$
 4. Si tras X no hay ningún símbolo, entonces $\text{siguiente}(Y)$ está en $\text{siguiente}(X)$

15

Conjuntos primero y siguiente

Ejemplos

- Dada la siguiente gramática

```
G=< {E,E',T,T',F},  
    {+,*,(,),id}  
    {  
      E  → TE'  
      E' → +TE' | λ  
      T  → FT'  
      T' → *FT' | λ  
      F  → (E)  | id    },  
E>  
  
primero(E)=primero(T)=primero(F)={(,id}  
primero(E')={+,λ}  
primero(T')={*,λ}  
siguiente(E)=siguiente(E')={),}$}  
siguiente(T)=siguiente(T')={+,),}$}  
siguiente(F)={+,*,),}$}
```

16

Conjuntos primero y siguiente

Ejemplos

- Dada la siguiente gramática

```
G=< {A,B,C,D},  
    {a,b,c,d}  
    {  
      A → BCD  
      B → b | BD  
      C → cA | a  
      D → λ | d  
    },  
    A>
```

```
primero(A)=primero(B)= {b}  
primero(C)={a,c}  
primero(D)={d,λ}  
siguiente(A)=siguiente(C)={d,$}  
siguiente(B)={a,c,d}  
siguiente(D)={a,c,d,$}
```

17

Conjuntos primero y siguiente

Ejemplos

- Dada la siguiente gramática

```
G=< {A,B,C,D},  
    {a,b,c,d}  
    {  
      A → BCD  
      B → b | BD | λ  
      C → cA | a  
      D → λ | d  
    },  
    A>
```

```
primero(A)={a,b,c,d}  
Primero(B)={b,d,λ}  
primero(C)={a,c}  
primero(D)={d,λ}  
siguiente(A)=siguiente(C)={d,$}  
siguiente(B)={a,c,d}  
siguiente(D)={a,c,d,$}
```

18

Conflictos LR(0)

- Cuando al construir la tabla de análisis LR(0) no se puede conseguir que sea determinista (porque en alguna casilla hay más de una acción) diremos que **la gramática no es LR(0)**
- En caso contrario diremos que la **gramática es LR(0)**.
- La existencia del no determinismo se llama **conflicto**
- Los conflictos pueden ser de los siguientes tipos
 - **Conflictos reducción / desplazamiento:**
 - En estos casos las acciones de la casilla son de estos dos tipos
 - Como ejemplo recuérdese el caso del último ejercicio
 - **Conflictos reducción / reducción:**
 - En estos casos las acciones son de reducción
 - El conflicto consiste en no poder determinar cuál de las reglas reducir

Conflictos

- Lo dicho antes para LR(0) puede aplicarse igual para todas las técnicas ascendentes que se estudiarán. Así
- Cuando al construir la tabla de análisis SLR(1), LR(1) o LALR(1) no se puede conseguir que sea determinista (porque en alguna casilla hay más de una acción) diremos, respectivamente, que **la gramática no es SLR(1), LR(1) o LALR(1)**

b.3

Construcción de la tabla

31

Estructura de la tabla de análisis

- Hay tantas filas como estados, y tantas columnas como símbolos hay en la gramática:
 - Los símbolos **terminales** determina la **acción** que realiza el analizador en cada situación.
 - Los símbolos **no terminales** determina la función **lr_a**.

ENTRADA

a_1
a_2
...
a_u
\$



TABLA ANÁLISIS

E	Σ_T			Σ_N		
	T_1		T_n	N_1		N_m
s_0						
...	
s_k						
Acción				lr_a		

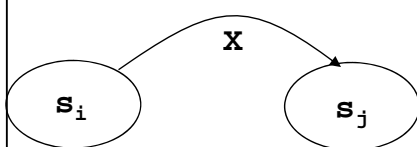
$G = \langle \Sigma_N, \Sigma_T, P, E \rangle$

32

- **La entrada** tiene inicialmente la cadena que se quiere analizar más un nuevo símbolo que indica su final y que históricamente es “\$”
- **La tabla**, contiene
 - **En su parte “acción”**, lo que hará el analizador cuando se encuentre en el estado de cada casilla y tenga como símbolo de lectura actual el terminal de la columna de la casilla y puede contener:
 - **ds**, desplazar el estado “s” (s es un estado del analizador)
 - **rP**, reducir mediante la regla de producción “P”
 - **aceptar**, cuando se termina el análisis aceptando
 - **error**, cuando se termina con error
 - **En su parte “ir_a”**, las transiciones entre los estados, por lo que sus casillas contienen estados del analizador.

Construcción de la tabla de análisis SLR(1)

- 1) Construir el **autómata finito determinista** que representa al analizador SLR(1).
- 2) Añadir tantas **filas** como estados tiene el autómata.
- 3) Añadir tantas **columnas** como símbolos terminales (acción) y no terminales (ir_a).
- 4) Leer las transiciones del autómata para insertar los desplazamientos en la tabla:
 - Si en el autómata se transita del estado s_i al s_j mediante el símbolo (terminal o no) x , entonces se añade a la casilla la acción



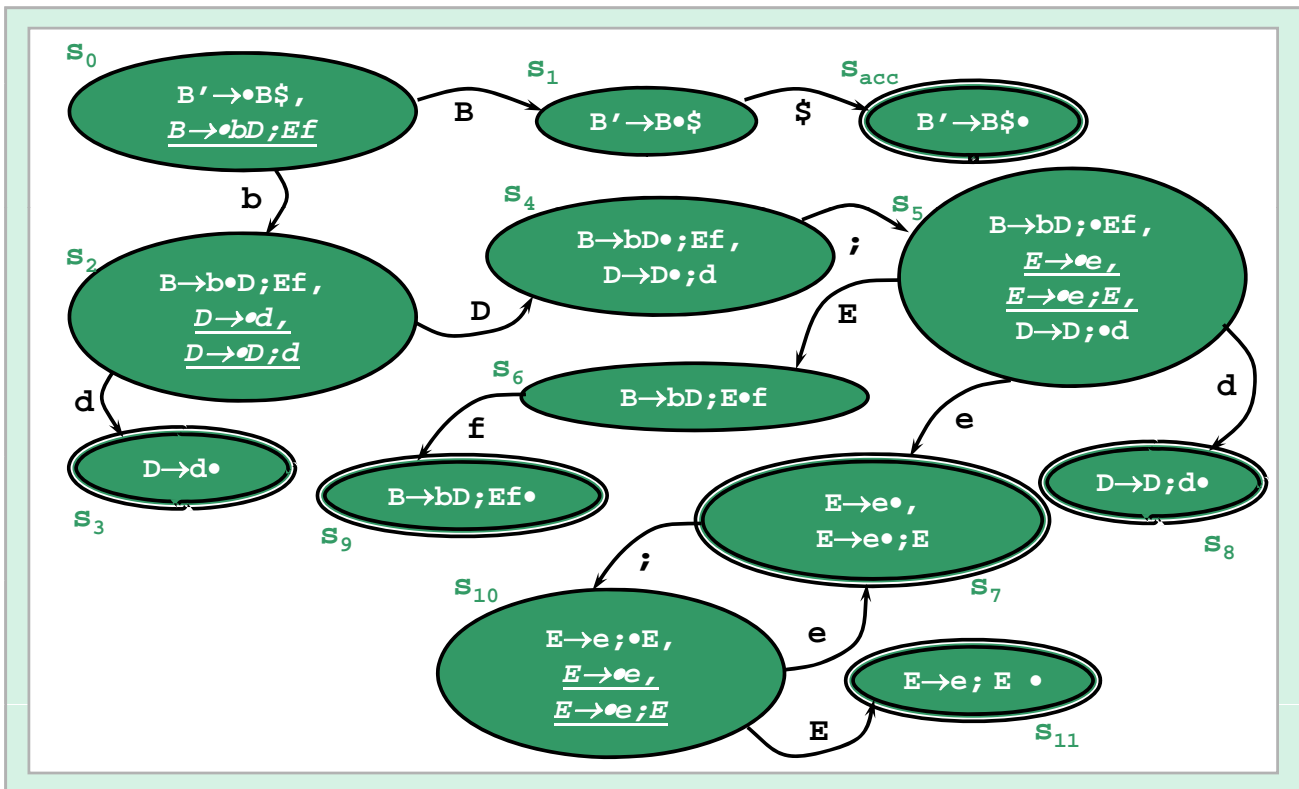
$$\text{Tabla_sintáctica}[i, X] = \begin{cases} dj & \text{si } x \in \Sigma_T \\ j & \text{si } x \in \Sigma_N \end{cases}$$

- 5) En las casillas a cada uno de los estados con configuraciones de reducción (por tanto contendrá una configuración del tipo $A \rightarrow \gamma \bullet$) hay que añadir la reducción de la regla $A \rightarrow \gamma$ **sólo en las columnas** correspondientes a los terminales de **siguiente(A)** que **son los que pueden seguir al no terminal de la parte izquierda de la regla**.
- 6) Si un estado s_i transita con el terminal \$ al estado que contiene la configuración de reducción de la regla añadida $\text{axioma}' \rightarrow \text{axioma} \$$. hay que añadir a $\text{Tabla_sintáctica}[i, \$]$ la acción **aceptar**.
- 7) Todas las demás casillas tienen asociada la acción de **error** y se suelen dejar en blanco.

Análisis LR(0)

Construcción de tablas de análisis LR(0): ejercicio

- Autómata finito determinista de transiciones:



35

Ejemplo de tabla de análisis SLR(1)

Gramática:

- (0) $B' \rightarrow B \$$
- (1) $B \rightarrow b D; E f$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D; d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e; E$

- $\text{siguientes}(B) = \{ \$ \}$
- $\text{siguientes}(D) = \{ ; \}$
- $\text{siguientes}(E) = \{ f \}$

	Σ_T						Σ_N		
E	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11					r5				
Acción							lr_a		

36

Conclusión

- El análisis SLR(1) es una versión mejorada del análisis LR(0) que
 - Comparte la técnica de creación de estados y, por tanto:
 - Emplea el mismo algoritmo para la operación **cierre**
 - Emplea el mismo algoritmo para la operación **ir_a**
 - Utiliza la tabla de análisis para tratar las entradas de la misma manera
 - Rellena las operaciones de desplazamiento de la misma manera
 - Pero tiene en cuenta que las reducciones sólo se produzcan en presencia de los terminales que pueden seguir al no terminal de la parte izquierda de la regla reducida

b.4

Uso de la tabla
(igual que LR(0))

Análisis SLR(1)

- Como ya se comentó en el tema anterior, siempre en el análisis sintáctico hay dos grandes fases: la construcción de la tabla y su uso. Los distintos algoritmos modifican la construcción de la tabla, pero su uso no cambia.
- Por lo tanto, el análisis SLR(1) es igual que el análisis LR(0) pero usando la tabla SLR(1) como la que acabamos de construir.
- Hagamos por lo tanto ahora un ejercicio que compruebe la corrección de dicha tabla analizando con ella las siguientes entradas (ignorar los saltos de línea y los blancos):

- Correcta:**

```
begin
  dec;
  ejec;
  ejec
end
```

- Incorrecta:**

```
begin
  dec;
  ejec;
end
```

Gramática:

```
(0) <Bloque'> → <Bloque>$
(1) <Bloque> → begin <Decs> ;
               <Ejecs> end
(2) <Decs> → dec
(3)         | <Decs>;dec
(4) <Ejecs> → ejec
(5)         | ejec ; <Ejecs>
```

Acrónimos:

```
<Bloque'>, B'
<Bloque>, B
<Decs>, D
<Ejecs>, E
begin, b
dec, d
end, f
ejec, e
```

39

Análisis SLR(1) : ejercicio

	Σ_T						Σ_N		
E	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11					r5				
	Acción						lr_a		

b d ; e ; e f \$

0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD;Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D;d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e;E$

40

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11				r5					
Acción							lr_a		

b d ; e ; e f \$

2 b 0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD; Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D; d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e; E$

41

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11				r5					
Acción							lr_a		

b d ; e ; e f \$

3 d 2 b 0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD; Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D; d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e; E$

42

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11					r5				
Acción							lr_a		

b d ; e ; e f \$

4 D 2 b 0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD; Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D; d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e; E$

43

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11					r5				
Acción							lr_a		

b d ; e ; e f \$

5 ; 4 D 2 b 0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD; Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D; d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e; E$

44

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3					r2				
4					d5				
5	d8	d7							6
6					d9				
7					d10	r4			
8					r3				
9						r1			
10		d7							11
11					r5				
Acción							lr_a		

b d ; e ; e f \$

7 e 5 ; 4 D 2 b 0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD; Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D;d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e; E$

45

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3					r2				
4					d5				
5	d8	d7							6
6					d9				
7					d10	r4			
8					r3				
9						r1			
10		d7							11
11					r5				
Acción							lr_a		

b d ; e ; e f \$

10 ; 7 e 5 ; 4 D 2 b 0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD; Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D;d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e; E$

46

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11					r5				
Acción							lr_a		

b	d	;	e	;	e	f	\$												
---	---	---	---	---	---	---	----	--	--	--	--	--	--	--	--	--	--	--	--

7	e	10	;	7	e	5	;	4	D	2									
b	0																		

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD;Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D;d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e;E$

47

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11					r5				
Acción							lr_a		

b	d	;	e	;	e	f	\$												
---	---	---	---	---	---	---	----	--	--	--	--	--	--	--	--	--	--	--	--

11	E	10	;	7	e	5	;	4	D	2									
b	0																		

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD;Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D;d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e;E$

48

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11				r5					
Acción							lr_a		

b d ; e ; e f \$

6 E 5 ; 4 D 2 b 0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD; Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D;d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e; E$

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11				r5					
Acción							lr_a		

b d ; e ; e f \$

9 f 6 E 5 ; 4 D 2 b 0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD; Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D;d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e; E$

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11					r5				
Acción							lr_a		

b d ; e ; e f \$

1 B 0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD;Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D;d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e;E$

51

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11					r5				
Acción							lr_a		

b d ; e ; e f \$

1 B 0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD;Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D;d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e;E$

52

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11				r5					
Acción							lr_a		

b d ; e ; \$

0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD; Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D; d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e; E$

53

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11				r5					
Acción							lr_a		

b d ; e ; \$

2 b 0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD; Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D; d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e; E$

54

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11				r5					
Acción							lr_a		

b d ; e ; \$

3 d 2 b 0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD; Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D; d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e; E$

55

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11				r5					
Acción							lr_a		

b d ; e ; \$

4 D 2 b 0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD; Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D; d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e; E$

56

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11					r5				
Acción							lr_a		

b d ; e ; \$

5 ; 4 D 2 b 0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD; Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D; d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e; E$

57

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11					r5				
Acción							lr_a		

b d ; e ; \$

7 e 5 ; 4 D 2 b 0

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD; Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D; d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e; E$

58

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11					r5				
Acción							lr_a		

b	d	;	e	;	\$									
---	---	---	---	---	----	--	--	--	--	--	--	--	--	--

▶

10	;	7	e	5	;	4	D	2	b	0
----	---	---	---	---	---	---	---	---	---	---

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD;Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D;d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e;E$

Análisis SLR(1) : ejercicio

E	Σ_T						Σ_N		
	d	e	b	;	f	\$	B	D	E
0			d2				1		
1						acc			
2	d3							4	
3				r2					
4				d5					
5	d8	d7							6
6					d9				
7				d10	r4				
8				r3					
9						r1			
10		d7							11
11					r5				
Acción							lr_a		

b	d	;	e	;	\$									
---	---	---	---	---	----	--	--	--	--	--	--	--	--	--

▶

10	;	7	e	5	;	4	D	2	b	0
----	---	---	---	---	---	---	---	---	---	---

- (0) $B' \rightarrow B\$$
- (1) $B \rightarrow bD;Ef$
- (2) $D \rightarrow d$
- (3) $D \rightarrow D;d$
- (4) $E \rightarrow e$
- (5) $E \rightarrow e;E$

b.5

Limitaciones de SLR(1)

61

Necesidad de otros algoritmos de análisis

Introducción

- Hay gramáticas interesantes para los lenguajes de programación de alto nivel que no son SLR(1)
- En estos casos se puede utilizar algoritmos más potentes
- Podemos mencionar, por ejemplo
 - LR(k), $k \geq 1$
 - LALR(1)

62

Necesidad de otros algoritmos de análisis

- Construya la tabla de análisis SLR(1) de la siguiente gramática (ambigua) que genera expresiones aritméticas

(1) $E \rightarrow E + E$

(2) $E \rightarrow E * E$

(3) $E \rightarrow i$

- Lo primero es aumentar la gramática

(0) $E' \rightarrow E \$$

(1) $E \rightarrow E + E$

(2) $E \rightarrow E * E$

(3) $E \rightarrow i$

63

Necesidad de otros algoritmos de análisis

Ejemplo

- Y la tabla de análisis SLR(1)

(0) $E' \rightarrow E \$$

(1) $E \rightarrow E + E$

(2) $E \rightarrow E * E$

(3) $E \rightarrow i$

$\text{primero}(E) = \{i\}$

$\text{siguiente}(E) = \{\$, +, *\}$

	Σ_T				
E	*	+	i	\$	B
0			s1		2
1	r3	r3		r3	
2	s4	s3		acc	
3			s1		6
4			s1		5
5	r2/s4	r2/s3		r2	
6	r1/s4	r1/s3		r1	
Acción					

- La gramática no es SLR(1)

65

Necesidad de otros algoritmos de análisis

Ejemplo: posibles soluciones

- Si el alumno reflexiona, el problema está originado por la ambigüedad ($i+i*i=(i+i)*i$? o $i+i*i=i+(i*i)$?)
- En este caso una posible solución es forzar una decisión en las casillas conflictivas.
- Hay que tener en cuenta la manera en la que afecta a la asignación de prioridades las diferentes alternativas.
- El alumno puede comprobar:

Necesidad de otros algoritmos de análisis

Ejemplo: posibles soluciones

- El efecto de desplazar en el caso de encontrar '+' y reducir en el caso de encontrar '*'

(0) $E' \rightarrow E\$$

(1) $E \rightarrow E+E$

(2) $E \rightarrow E * E$

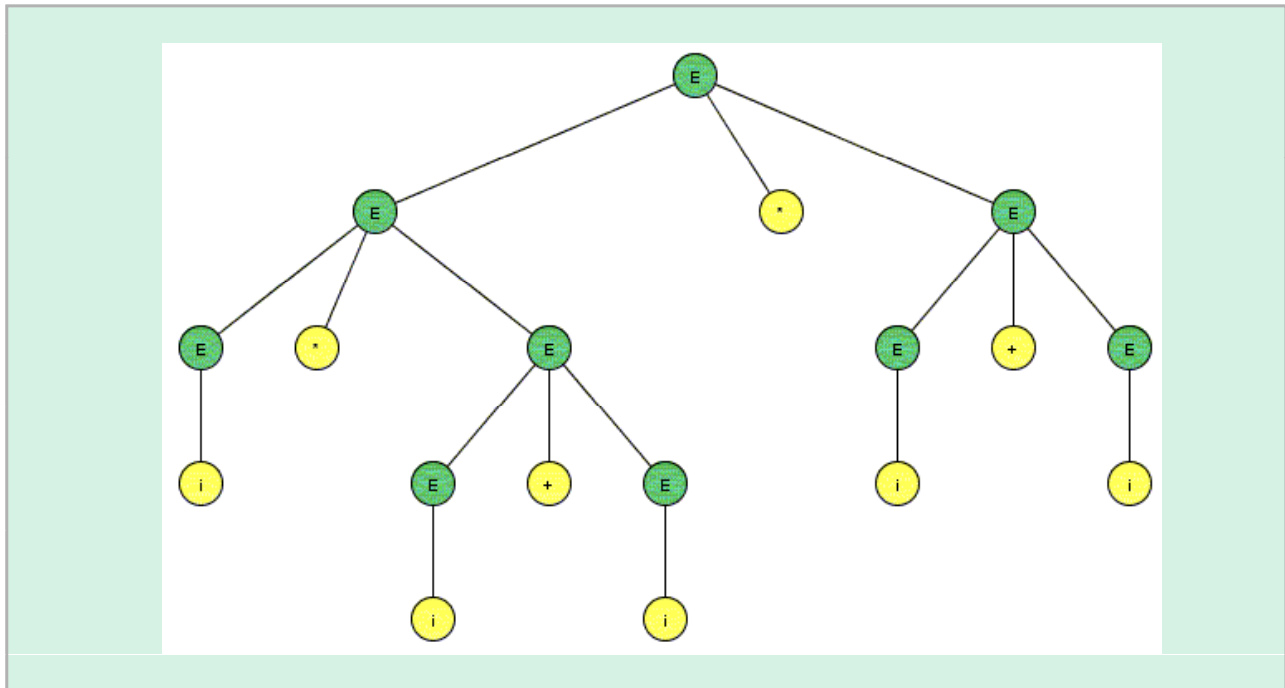
(3) $E \rightarrow i$

	Σ_T				
E	*	+	i	\$	B
0			s1		2
1	r3	r3		r3	
2	s4	s3		acc	
3			s1		6
4			s1		5
5	r2	s3		r2	
6	r1	s3		r1	
	Acción				

Necesidad de otros algoritmos de análisis

Ejemplo: posibles soluciones

- El análisis de la cadena 'i*i+i*i+i'



68

Necesidad de otros algoritmos de análisis

Ejemplo: posibles soluciones

- El efecto de desplazar en el caso de encontrar '*' y reducir en el caso de encontrar '+'

(0) $E' \rightarrow E\$$

(1) $E \rightarrow E+E$

(2) $E \rightarrow E * E$

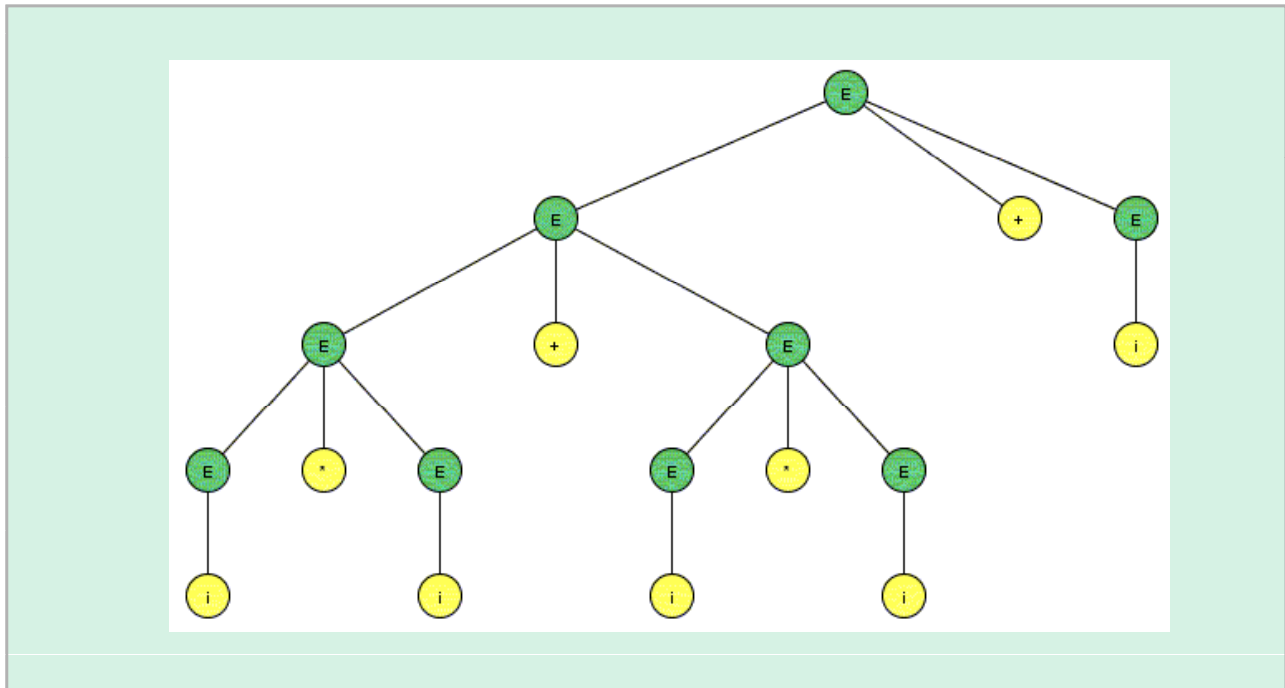
(3) $E \rightarrow i$

	Σ_T				
E	*	+	i	\$	B
0			s1		2
1	r3	r3		r3	
2	s4	s3		acc	
3			s1		6
4			s1		5
5	s4	r2		r2	
6	s4	r1		r1	
	Acción				

69

Ejemplo: posibles soluciones

- El análisis de la cadena 'i*i+i*i+i'



b.6

Ejercicios de repaso

Ejercicio de repaso 1

- Construya la tabla de análisis SLR(1) de la siguiente gramática que genera el lenguaje

$$\{xb, a^nx b^n \mid n \geq 0\}$$

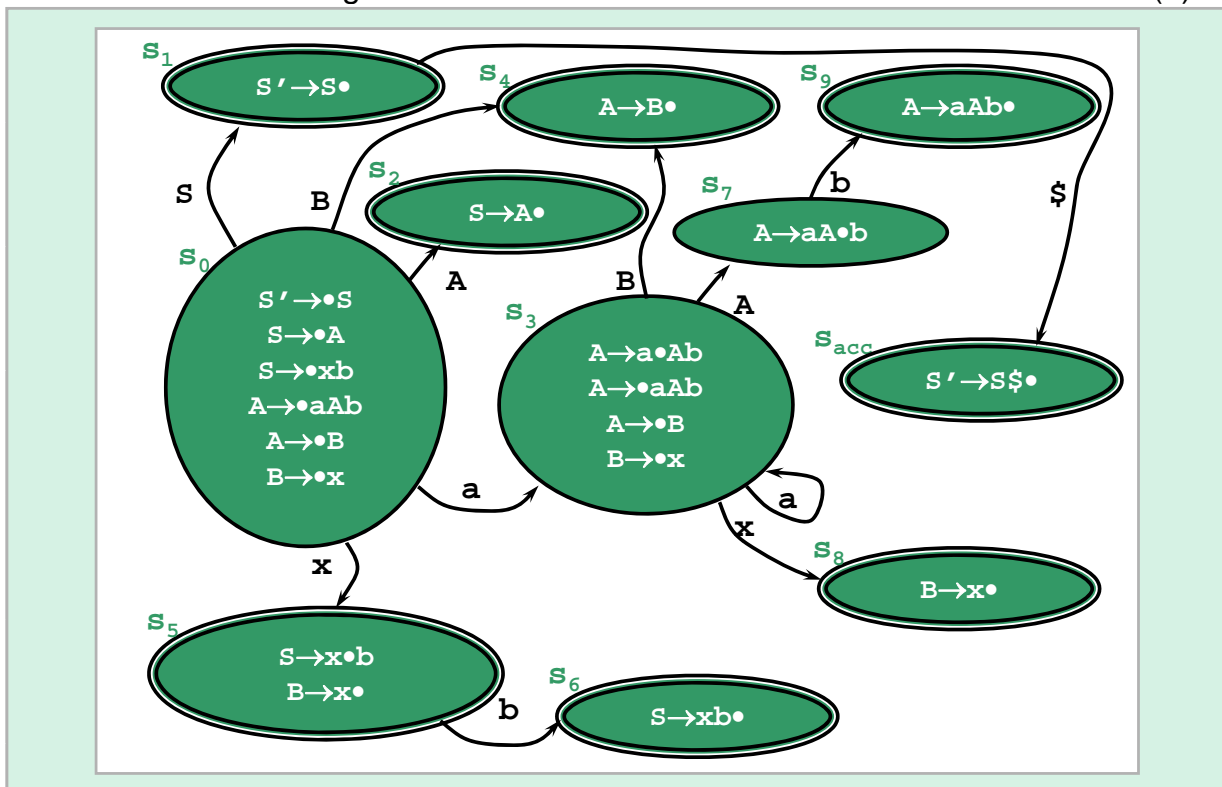
- (1) $S \rightarrow A$
- (2) $S \rightarrow xb$
- (3) $A \rightarrow aAb$
- (4) $A \rightarrow B$
- (5) $B \rightarrow x$

- Lo primero es aumentar la gramática

- (0) $S' \rightarrow S\$$
- (1) $S \rightarrow A$
- (2) $S \rightarrow xb$
- (3) $A \rightarrow aAb$
- (4) $A \rightarrow B$
- (5) $B \rightarrow x$

Ejercicio de repaso 1

- Calculamos el diagrama de transiciones entre estados del autómata SLR(1)



Ejercicio de repaso 1

- Y la tabla de análisis SLR(1)

(0) $S' \rightarrow S\$$

(1) $S \rightarrow A$

(2) $S \rightarrow xb$

(3) $A \rightarrow aAb$

(4) $A \rightarrow B$

(5) $B \rightarrow x$

$\text{primero}(A) = \{a, x\}$

$\text{primero}(B) = \{x\}$

$\text{primero}(S) = \{a, x\}$

$\text{siguiente}(A) = \{\$, b\}$

$\text{siguiente}(B) = \{\$, b\}$

$\text{siguiente}(S) = \{\$\}$

	Σ_T				Σ_N		
E	a	b	x	\$	S	A	B
0	s3		s5		1	2	4
1				acc			
2				r1			
3	s3		s8			7	4
4		r4		r4			
5		r5/s6		r5			
6				r2			
7		s9					
8		r5		r5			
9		r3		r3			
	Acción				lr_a		

74

Repaso

Glosario

- LR(k):**
 - Técnica de análisis sintáctico ascendente que:
 - Examina la entrada de izquierda a derecha (*left-to-right*)
 - Construye una derivación derecha de la palabra analizada (*right-most derivation*)
 - Utiliza para ello los k siguientes símbolos (tokens) de la entrada
- SLR:**
 - Técnica de análisis sintáctico ascendente que es una simplificación de LR(k).
 - Toma su nombre de la expresión inglesa que significa LR sencillo
- LALR:**
 - Técnica de análisis sintáctico ascendente que es una extensión de la técnica LR pero utilizando símbolos de anticipación.
 - Toma su nombre de la expresión inglesa *look-ahead LR* que significa precisamente eso.

75