

Examen 3, Compiladores

Bernal Núñez Raúl
Cureño Sanchez Misael

Fecha límite de entrega: Miercoles 08 de noviembre a las 23:59.

1. (25 pts.) Dada la siguiente gramática:

$$\begin{aligned} E &\rightarrow (E + T) \mid id \\ T &\rightarrow (T * F) \mid id \\ F &\rightarrow id \end{aligned}$$

- a) Construir el *AFD* y la tabla de análisis para un analizador *LR(0)*.

Solución:

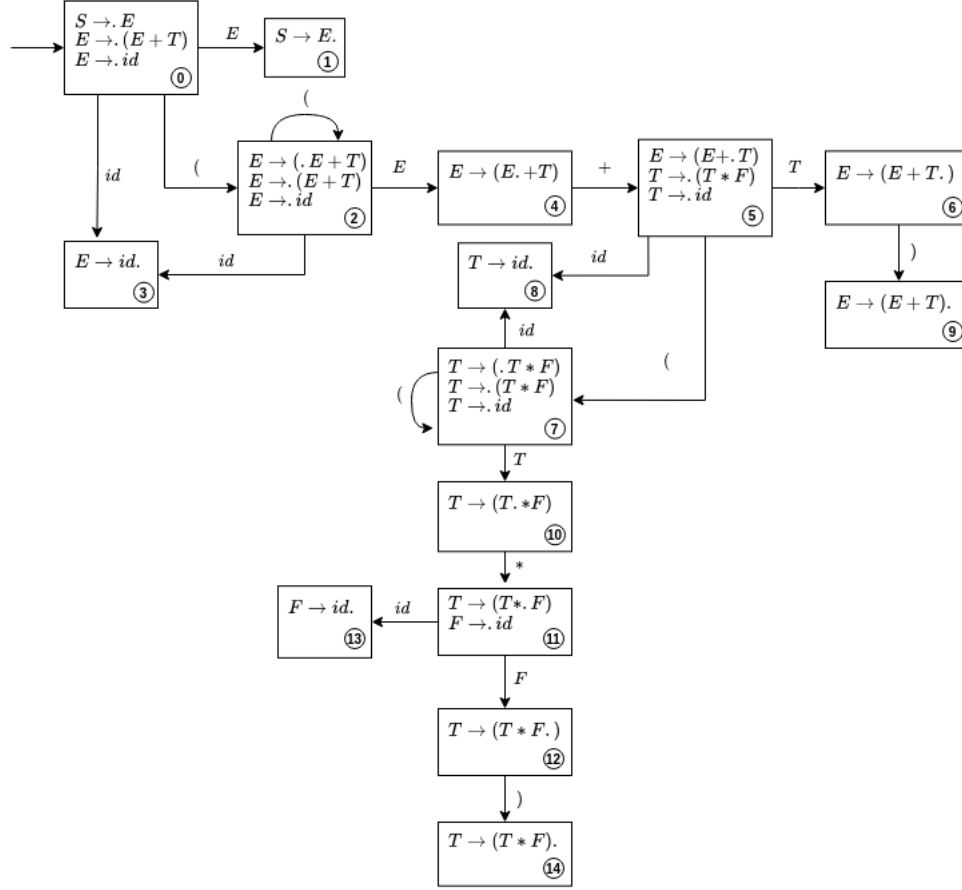
Primero es necesario modificar la gramática para agregar una nueva producción: $S \rightarrow E$, donde S es nuestro símbolo inicial.

$$\begin{aligned} S &\rightarrow E \\ E &\rightarrow (E + T) \mid id \\ T &\rightarrow (T * F) \mid id \\ F &\rightarrow id \end{aligned}$$

Tenemos los siguientes elementos *LR(0)*:

$$\begin{aligned} S &\rightarrow .E \\ S &\rightarrow E. \\ E &\rightarrow .(E + T) \\ E &\rightarrow (.E + T) \\ E &\rightarrow (E. + T) \\ E &\rightarrow (E + .T) \\ E &\rightarrow (E + T.) \\ E &\rightarrow (E + T). \\ E &\rightarrow .id \\ E &\rightarrow id. \\ T &\rightarrow .(T * F) \\ T &\rightarrow (.T * F) \\ T &\rightarrow (T. * F) \\ T &\rightarrow (T * .F) \\ T &\rightarrow (T * F.) \\ T &\rightarrow (T * F). \\ T &\rightarrow .id \\ T &\rightarrow id. \\ F &\rightarrow .id \\ F &\rightarrow id. \end{aligned}$$

Ahora vamos a construir directamente el AFD:



La idea es que empezamos con el primer estado que tendrá las producciones de S y E donde el punto está al principio como podemos observar en el estado 0. Asimismo, de cada estado se van a generar las transiciones a partir del símbolo que esté después del punto en las respectivas producciones del estado, como en el estado 0 donde tiene tres producciones y se generan tres transiciones con E , $($ e id que son los terminales o no terminales a la derecha del punto.

Finalmente, el nuevo estado que se genera a través de dichas transiciones siempre va a contener la respectiva producción en la cual el punto se movió un símbolo a la derecha, por ejemplo, estando en el estado 0 si vemos la segunda producción $E \rightarrow .(E + T)$, se genera la transición con $($ y el nuevo estado, en este caso 2, tendrá la producción donde el punto aparece después de $($, es decir, $E \rightarrow (.E + T)$.

Sin embargo, tenemos que verificar que si el nuevo estado generado con la producción que difiere únicamente en el punto recorrido a la derecha, tiene el punto antes de un no terminal, entonces tenemos que agregar al estado todas las producciones de dicho no terminal, pero donde el punto está al principio. Lo anterior lo podemos notar con el estado 2 el cual se genera con $E \rightarrow (.E + T)$, pero después del punto está el no terminal E provocando que se tengan que agregar las producciones de E que tienen el punto al principio al estado 2.

Siguiendo las ideas anteriores, nos queda el AFD y con eso ya podemos generar nuestra

tabla de análisis sintáctico para un analizador $LR(0)$, donde las transiciones nos ayudarán a completar las acciones de desplazamiento y las columnas de los no terminales para saber a qué estado ir. Asimismo, los estados que tienen un elemento completo nos ayudarán a completar las acciones de reducción:

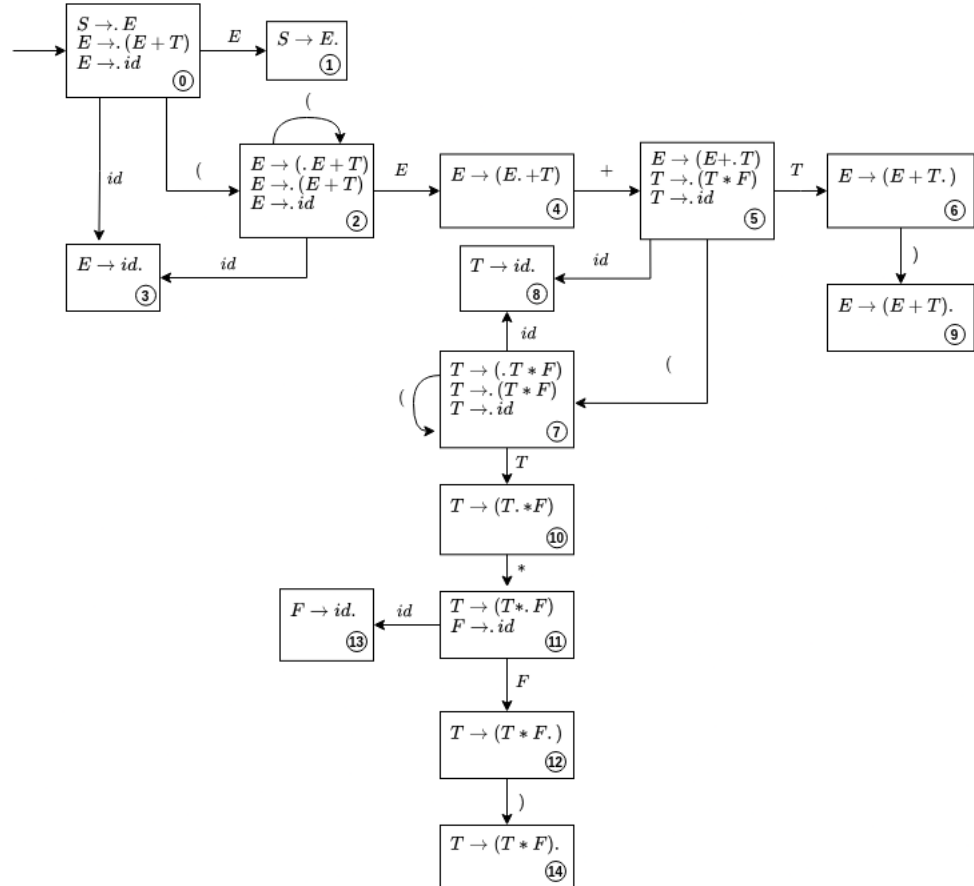
	()	+	*	<i>id</i>	\$	<i>E</i>	<i>T</i>	<i>F</i>
0	$D-2$				$D-3$		1		
1						Aceptar			
2	$D-2$				$D-3$		4		
3	$R: E \rightarrow id$	$R: E \rightarrow id$	$R: E \rightarrow id$	$R: E \rightarrow id$	$R: E \rightarrow id$	$R: E \rightarrow id$			
4			$D-5$						
5	$D-7$				$D-8$			6	
6		$D-9$							
7	$D-7$				$D-8$			10	
8	$R: T \rightarrow id$	$R: T \rightarrow id$	$R: T \rightarrow id$	$R: T \rightarrow id$	$R: T \rightarrow id$	$R: T \rightarrow id$			
9	$R: E \rightarrow (E+T)$	$R: E \rightarrow (E+T)$	$R: E \rightarrow (E+T)$	$R: E \rightarrow (E+T)$	$R: E \rightarrow (E+T)$	$R: E \rightarrow (E+T)$			
10				$D-11$					
11					$D-13$				12
12		$D-14$							
13	$R: F \rightarrow id$	$R: F \rightarrow id$	$R: F \rightarrow id$	$R: F \rightarrow id$	$R: F \rightarrow id$	$R: F \rightarrow id$			
14	$R: T \rightarrow (T * F)$	$R: T \rightarrow (T * F)$	$R: T \rightarrow (T * F)$	$R: T \rightarrow (T * F)$	$R: T \rightarrow (T * F)$	$R: T \rightarrow (T * F)$			

Finalmente, podemos ver que la gramática proporcionada es $LR(0)$ pues se logró generar la tabla de análisis sintáctica de manera correcta sin conflictos. □

b) Construir el AFD y la tabla de análisis para un analizador $SLR(1)$.

Solución:

Para este inciso podemos reutilizar el autómata que generamos para el analizador $LR(0)$:



La diferencia es que ahora necesitaremos calcular los conjuntos Siguiente de la gramática:

$$\begin{aligned} \text{Siguiente}(S) &= \{\$ \} \\ \text{Siguiente}(E) &= \{\$, + \} \\ \text{Siguiente}(T) &= \{), * \} \\ \text{Siguiente}(F) &= \{ \} \end{aligned}$$

Con ayuda del autómata y de los conjuntos Siguiente generamos la tabla de análisis sintáctico, donde se realizará lo mismo que para $LR(0)$ con la diferencia de que las casillas en las que se añadirá la producción que nos indica cómo reducir será únicamente en las columnas correspondientes a los respectivos elementos de los conjuntos Siguiente:

	()	+	*	<i>id</i>	\$	<i>E</i>	<i>T</i>	<i>F</i>
0	$D-2$				$D-3$		1		
1						<i>Aceptar</i>			
2	$D-2$				$D-3$		4		
3			$R: E \rightarrow id$			$R: E \rightarrow id$			
4			$D-5$						
5	$D-7$				$D-8$			6	
6		$D-9$							
7	$D-7$				$D-8$			10	
8		$R: T \rightarrow id$		$R: T \rightarrow id$					
9			$R: E \rightarrow (E+T)$			$R: E \rightarrow (E+T)$			
10				$D-11$					
11					$D-13$				12
12		$D-14$							
13		$R: F \rightarrow id$							
14		$R: T \rightarrow (T * F)$		$R: T \rightarrow (T * F)$					

Finalmente, podemos ver que la gramática proporcionada es también $SLR(1)$ pues se logró generar la tabla de análisis sintáctica de manera correcta sin conflictos. \square

- c) Reconocer la cadena $(6 + (4 * 2))$ con ambos analizadores.

Solución:

Utilizando nuestro analizador $LR(0)$:

Paso	Pila de análisis sintáctico	Entrada	Acción a aplicar
1	\$ 0	$(6 + (4 * 2))\$$	$D-2$
2	\$ 0 (2	$6 + (4 * 2))\$$	$D-3$
3	\$ 0 (2 6 3	$+(4 * 2))\$$	$R: E \rightarrow id$
4	\$ 0 (2 E 4	$+(4 * 2))\$$	$D-5$
5	\$ 0 (2 E 4 + 5	$(4 * 2))\$$	$D-7$
6	\$ 0 (2 E 4 + 5 (7	$4 * 2))\$$	$D-8$
7	\$ 0 (2 E 4 + 5 (7 4 8	$* 2))\$$	$R: T \rightarrow id$
8	\$ 0 (2 E 4 + 5 (7 T 10	$* 2))\$$	$D-11$
9	\$ 0 (2 E 4 + 5 (7 T 10 * 11	$2))\$$	$D-13$
10	\$ 0 (2 E 4 + 5 (7 T 10 * 11 2 13	$))\$$	$R: F \rightarrow id$
10	\$ 0 (2 E 4 + 5 (7 T 10 * 11 F 12	$))\$$	$D-14$
10	\$ 0 (2 E 4 + 5 (7 T 10 * 11 F 12) 14	$)\$$	$R: T \rightarrow (T * F)$
10	\$ 0 (2 E 4 + 5 T 6	$)\$$	$D-9$
10	\$ 0 (2 E 4 + 5 T 6) 9	$\$$	$R: E \rightarrow (E + T)$
10	\$ 0 E 1	$\$$	<i>Aceptar</i>

Utilizando ahora nuestro analizador $SLR(1)$:

Paso	Pila de análisis sintáctico	Entrada	Acción a aplicar
1	\$ 0	$(6 + (4 * 2))\$$	$D - 2$
2	\$ 0 (2	$6 + (4 * 2))\$$	$D - 3$
3	\$ 0 (2 6 3	$+(4 * 2))\$$	$R : E \rightarrow id$
4	\$ 0 (2 E 4	$+(4 * 2))\$$	$D - 5$
5	\$ 0 (2 E 4 + 5	$(4 * 2))\$$	$D - 7$
6	\$ 0 (2 E 4 + 5 (7	$4 * 2))\$$	$D - 8$
7	\$ 0 (2 E 4 + 5 (7 4 8	$* 2))\$$	$R : T \rightarrow id$
8	\$ 0 (2 E 4 + 5 (7 T 10	$* 2))\$$	$D - 11$
9	\$ 0 (2 E 4 + 5 (7 T 10 * 11	$2))\$$	$D - 13$
10	\$ 0 (2 E 4 + 5 (7 T 10 * 11 2 13	$)\$$	$R : F \rightarrow id$
10	\$ 0 (2 E 4 + 5 (7 T 10 * 11 F 12	$)\$$	$D - 14$
10	\$ 0 (2 E 4 + 5 (7 T 10 * 11 F 12) 14	$)\$$	$R : T \rightarrow (T * F)$
10	\$ 0 (2 E 4 + 5 T 6	$)\$$	$D - 9$
10	\$ 0 (2 E 4 + 5 T 6) 9	$\$$	$R : E \rightarrow (E + T)$
10	\$ 0 E 1	$\$$	<i>Aceptar</i>

Con lo anterior, podemos notar que ambos analizadores aceptan la cadena correspondiente de la misma manera, en este caso no vemos un cambio porque la cadena es válida y además desde la tabla que se utilizó para $LR(0)$ no hubo conflictos, por lo que al hacer la tabla de $SLR(1)$ tampoco existieron conflictos. Sin embargo, si habrá algunos casos cuando una cadena no sea válida donde el analizador $SLR(1)$ se dará cuenta antes de un error.

□

2. (25 pts.) Indica si los siguientes enunciados son falsos o verdaderos. Justifica tu respuesta.

a) Una gramática $LR(1)$ que no es $LALR(1)$ debe tener sólo conflictos de reducción-reducción.

Solución:

Si tenemos una gramática $LR(1)$ que no es $LALR(1)$, entonces únicamente debería tener conflictos de desplazamiento-reducción, pues los conflictos de reducción-reducción pueden surgir al momento de simplificar el autómata de $LR(1)$ al autómata de $LALR(1)$, sin embargo, estamos suponiendo que $LR(1)$ no es $LALR(1)$, así dicha simplificación no se da, ocasionando que únicamente se tengan los conflictos de desplazamiento-reducción. Por lo tanto, es falso que una gramática $LR(1)$ que no es $LALR(1)$ debe tener sólo conflictos de reducción-reducción.

□

b) Pueden haber gramáticas $SLR(1)$ que no sean $LALR(1)$.

Solución:

Considero que no pueden haber gramáticas $SLR(1)$ que no sean $LALR(1)$, pues supongamos que G_1 y G_2 son gramáticas reconocidas por el autómata de $SLR(1)$ y de $LALR(1)$, respectivamente, entonces tenemos que para generar el autómata de $LALR(1)$ se utiliza en los elementos de cada estado un elemento $LR(0)$ junto con no terminales o $\$$, es decir, usa un elemento $LR(1)$ con la finalidad de que se pueda basar en un token de la entrada para decidir si seguir con el análisis de la cadena o rechazarla. Asimismo, en el autómata de $SLR(1)$ se tienen únicamente los elementos $LR(0)$, y en este caso se ayuda de los conjuntos Siguiendo y de un token de la entrada para saber si al momento de reducir, para así evitar realizar una reducción si el token de la entrada no corresponde

con lo esperado y así rechazar la cadena de entrada.

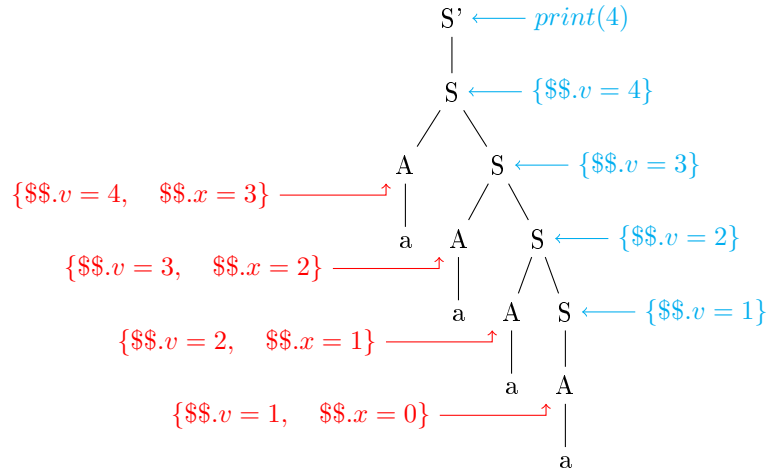
Con lo anterior podemos notar que el autómata de $SLR(1)$ lo que hace es evitar mayor número de conflictos con respecto a $LR(0)$, pues puede saber si rechazar una cadena antes de hacer una reducción, para que si no se puede rechace las cadenas no válidas más rápido, pero el autómata de $LALR(1)$ además de verificar si es válido hacer una reducción basado en la entrada y el elemento de un estado, también verifica si es válido realizar un desplazamiento utilizando igualmente el elemento de un estado y un token de la entrada. Por lo tanto, observamos que el autómata de $LALR(1)$ es más poderoso, con lo cual no puede haber una gramática $SLR(1)$ que no sea $LALR(1)$, entonces el enunciado es falso. \square

3. (25 pts) Dada la siguiente gramática con atributos:

$S' \rightarrow S$	$print(\$1.v)$
$S \rightarrow AS$	$\$$.v = \$1.v$ $\$1.x = \$2.v$
$S \rightarrow A$	$\$1.v = 0$ $\$$.x = \$1.v$
$A \rightarrow a$	$\$$.v = \$$.x + 1$

- a) Mostrar el árbol sintáctico (gramatical o *ASA*) decorado para la cadena de entrada *aaaa* e indicar la salida que provoca el análisis semántico.

Solución:



La salida es 4. \square

4. (25 pts) Dada la siguiente gramática que genera números binarios:

$$\begin{aligned} A &\rightarrow AB \mid B \\ B &\rightarrow 0 \mid 1 \end{aligned}$$

- a) Obtener la gramática con atributos que al analizar un número binario calcule su equivalente en decimal.

Solución:

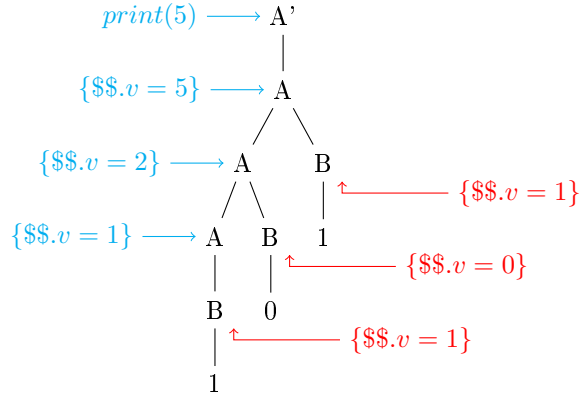
□

$A' \rightarrow A$	$print(\$$.v)$
$A \rightarrow AB$	$\$$.v = 2 * (\$1.v) + \$2.v$
$A \rightarrow B$	$\$$.v = \$1.v$
$B \rightarrow 0 \mid 1$	$\$$.v = 0 \mid 1$

- b) Mostrar el árbol para la cadena de entrada 101 e indicar la salida que provoca el análisis semántico.

Solución:

Árbol de sintaxis abstracta:



La salida es 5.

□