

**SOLUCIONES****Parte I: PREGUNTAS TIPO TEST. 30%.**

- | | | | | |
|-------|-------|-------|--------|--------|
| 1. b) | 4. a) | 7. c) | 10. a) | 13. c) |
| 2. b) | 5. c) | 8. c) | 11. c) | 14. c) |
| 3. c) | 6. c) | 9. a) | 12. b) | 15. c) |

Parte II: PREGUNTAS CORTAS. 10%.

Obtenemos los conjuntos PRIMERO y SIGUIENTE de los no terminales:

$\text{PRIMERO}(S) = \{ pa, es, \lambda \}$	$\text{SIGUIENTE}(S) = \{ \$ \}$
$\text{PRIMERO}(INF) = \{ es \}$	$\text{SIGUIENTE}(INF) = \{ pa, es, \$ \}$
$\text{PRIMERO}(COO) = \{ lat \}$	$\text{SIGUIENTE}(COO) = \{ zo \}$
$\text{PRIMERO}(COL) = \{ es, \lambda \}$	$\text{SIGUIENTE}(COL) = \{ pa, es, \$ \}$

Ahora obtenemos los conjuntos PREDICT de cada regla, y los contrastamos con la tabla de análisis del enunciado:

$\text{PREDICT}(S \rightarrow pa S) = \{ pa \}$	Correcto.
$\text{PREDICT}(S \rightarrow INF S) = \{ es \}$	Falta la regla $S \rightarrow INF S$ en la entrada $[S, es]$ de la tabla.
$\text{PREDICT}(S \rightarrow \lambda) = \{ \$ \}$	Correcto.
$\text{PREDICT}(INF \rightarrow es COO zo COL) = \{ es \}$	Sobra la regla $COO \rightarrow lat lon$ de la entrada $[INF, lat]$.
$\text{PREDICT}(COO \rightarrow lat lon) = \{ lat \}$	Correcto.
$\text{PREDICT}(COL \rightarrow es COL) = \{ es \}$	Correcto.
$\text{PREDICT}(COL \rightarrow \lambda) = \{ pa, es, \$ \}$	Falta la regla $COL \rightarrow \lambda$ en las entradas $[COL, es]$ y $[COL, \$]$.

Hay un error más en la tabla: no debe aparecer una columna para el símbolo COO , puesto que no es un terminal.

Parte III: PROBLEMA. 60%.**Apartado 1.**

Obtenemos los conjuntos PRIMERO y SIGUIENTE que solicita el enunciado:

$\text{PRIMERO}(E) = \{ mo, true, false \}$	$\text{SIGUIENTE}(E) = \{ and, \$, ,, \} \}$
$\text{PRIMERO}(L) = \{ mo, true, false \}$	$\text{SIGUIENTE}(L) = \{) \}$

La gramática del enunciado no puede ser una gramática LL(1) por tres razones:

1. Es ambigua por la regla $E \rightarrow E \text{ and } E$, ya que esta regla no fija la asociatividad del operador **and**. Podemos verlo obteniendo dos derivaciones más a la derecha (o dos árboles) de la entrada **true and true and true**:

$ \begin{aligned} E &\Rightarrow E \text{ and } E \\ &\Rightarrow E \text{ and } E \text{ and } E \\ &\Rightarrow E \text{ and } E \text{ and } true \\ &\Rightarrow E \text{ and } true \text{ and } true \\ &\Rightarrow true \text{ and } true \text{ and } true \end{aligned} $		$ \begin{aligned} E &\Rightarrow E \text{ and } E \\ &\Rightarrow E \text{ and } true \\ &\Rightarrow E \text{ and } E \text{ and } true \\ &\Rightarrow E \text{ and } true \text{ and } true \\ &\Rightarrow true \text{ and } true \text{ and } true \end{aligned} $	
---	--	---	--

2. Es recursiva por la izquierda con el no terminal E y la regla $E \rightarrow E \text{ and } E$.
3. No está factorizada, ya que las reglas $L \rightarrow E$, $L \mid E$ tienen el factor común E .

Apartado 2.

Se solicita calcular una gramática equivalente a la propuesta en el enunciado que sea LL(1).

El primer paso consiste en eliminar la ambigüedad de la gramática, y para ello se aplica un procedimiento similar al visto en la teoría de la asignatura para resolver la ambigüedad de la gramática de operaciones aritméticas. Como el enunciado indica que el operador **and** es asociativo por la izquierda, creamos un nuevo no terminal T para representar al operando derecho del operador **and**, o bien a una expresión sin el **and** como operador de primer nivel:

$$\begin{aligned} E &\rightarrow E \text{ and } T \mid T \\ T &\rightarrow \text{mo}(L) \mid \text{true} \mid \text{false} \\ L &\rightarrow E \text{ , } L \mid E \end{aligned}$$

El siguiente paso consiste en la eliminación de la recursión por la izquierda inmediata del no terminal E :

$$\begin{aligned} E &\rightarrow T \mid T E' \\ E' &\rightarrow \text{and } T \mid \text{and } T E' \\ T &\rightarrow \text{mo}(L) \mid \text{true} \mid \text{false} \\ L &\rightarrow E \text{ , } L \mid E \end{aligned}$$

Ahora se pueden eliminar los factores comunes de E , E' y L :

$$\begin{aligned} E &\rightarrow T E'' \\ E' &\rightarrow \text{and } T E'' \\ E'' &\rightarrow \lambda \mid E' \\ T &\rightarrow \text{mo}(L) \mid \text{true} \mid \text{false} \\ L &\rightarrow E L' \\ L' &\rightarrow \text{ , } L \mid \lambda \end{aligned}$$

Aunque no es un paso necesario, se puede simplificar un poco la gramática para obtener G' :

$$\begin{aligned} E &\rightarrow T E' \\ E' &\rightarrow \lambda \mid \text{and } T E' \\ T &\rightarrow \text{mo}(L) \mid \text{true} \mid \text{false} \\ L &\rightarrow E L' \\ L' &\rightarrow \text{ , } L \mid \lambda \end{aligned}$$

A partir de la última gramática, calculamos los conjuntos PRIMERO y SIGUIENTE:

$\text{PRIMERO}(E) = \{ \text{mo}, \text{true}, \text{false} \}$	$\text{SIGUIENTE}(E) = \{ \$, \text{ , }, \text{) } \}$
$\text{PRIMERO}(E') = \{ \lambda, \text{and} \}$	$\text{SIGUIENTE}(E') = \{ \$, \text{ , }, \text{) } \}$
$\text{PRIMERO}(T) = \{ \text{mo}, \text{true}, \text{false} \}$	$\text{SIGUIENTE}(T) = \{ \text{and}, \$, \text{ , }, \text{) } \}$
$\text{PRIMERO}(L) = \{ \text{mo}, \text{true}, \text{false} \}$	$\text{SIGUIENTE}(L) = \{ \text{) } \}$
$\text{PRIMERO}(L') = \{ \text{ , }, \lambda \}$	$\text{SIGUIENTE}(L') = \{ \text{) } \}$

Con los conjuntos anteriores podemos calcular los PREDICT de las reglas de la gramática G' :

$\text{PREDICT}(E \rightarrow T E') = \{ \text{mo}, \text{true}, \text{false} \}$	$\text{PREDICT}(T \rightarrow \text{false}) = \{ \text{false} \}$
$\text{PREDICT}(E' \rightarrow \lambda) = \{ \$, \text{ , }, \text{) } \}$	$\text{PREDICT}(L \rightarrow E L') = \{ \text{mo}, \text{true}, \text{false} \}$
$\text{PREDICT}(E' \rightarrow \text{and } T E') = \{ \text{and} \}$	$\text{PREDICT}(L' \rightarrow \text{ , } L) = \{ \text{ , } \}$
$\text{PREDICT}(T \rightarrow \text{mo}(L)) = \{ \text{mo} \}$	$\text{PREDICT}(L' \rightarrow \lambda) = \{ \text{) } \}$
$\text{PREDICT}(T \rightarrow \text{true}) = \{ \text{true} \}$	

Se puede observar que la intersección de los conjuntos PREDICT de cualquier par de reglas de un mismo no terminal es siempre vacía. Por tanto, la gramática equivalente G' es LL(1).

Apartado 3.

Los conjuntos de ítems de la colección LR(0) son los siguientes:

$$\begin{aligned}
I_0 &= \{ E' \rightarrow \cdot E \\
&\quad E \rightarrow \cdot E \text{ and } E \\
&\quad E \rightarrow \cdot \text{mo} (L) \\
&\quad E \rightarrow \cdot \text{true} \\
&\quad E \rightarrow \cdot \text{false} \} \\
I_1 &= \text{GOTO}(I_0, E) = \{ E' \rightarrow E \cdot \\
&\quad E \rightarrow E \cdot \text{ and } E \} \\
I_2 &= \text{GOTO}(I_0, \text{mo}) = \{ E \rightarrow \text{mo} \cdot (L) \} \\
I_3 &= \text{GOTO}(I_0, \text{true}) = \{ E \rightarrow \text{true} \cdot \} \\
I_4 &= \text{GOTO}(I_0, \text{false}) = \{ E \rightarrow \text{false} \cdot \} \\
I_5 &= \text{GOTO}(I_1, \text{and}) = \{ E \rightarrow E \text{ and } \cdot E \\
&\quad E \rightarrow \cdot E \text{ and } E \\
&\quad E \rightarrow \cdot \text{mo} (L) \\
&\quad E \rightarrow \cdot \text{true} \\
&\quad E \rightarrow \cdot \text{false} \} \\
I_6 &= \text{GOTO}(I_2, () = \{ E \rightarrow \text{mo} (\cdot L) \\
&\quad L \rightarrow \cdot E , L \\
&\quad L \rightarrow \cdot E \\
&\quad E \rightarrow \cdot E \text{ and } E \\
&\quad E \rightarrow \cdot \text{mo} (L) \\
&\quad E \rightarrow \cdot \text{true} \\
&\quad E \rightarrow \cdot \text{false} \} \\
I_7 &= \text{GOTO}(I_5, E) = \{ E \rightarrow E \text{ and } E \cdot \\
&\quad E \rightarrow E \cdot \text{ and } E \} \\
\text{GOTO}(I_5, \text{mo}) &= I_2 \\
\text{GOTO}(I_5, \text{true}) &= I_3 \\
\text{GOTO}(I_5, \text{false}) &= I_4 \\
\text{GOTO}(I_5, \text{and}) &= I_5 \\
\text{GOTO}(I_5, \text{mo}) &= I_2 \\
\text{GOTO}(I_5, \text{true}) &= I_3 \\
\text{GOTO}(I_5, \text{false}) &= I_4 \\
\text{GOTO}(I_5, \text{and}) &= I_5 \\
I_{10} &= \text{GOTO}(I_8,) = \{ E \rightarrow \text{mo} (L) \cdot \} \\
I_{11} &= \text{GOTO}(I_9, ,) = \{ L \rightarrow E , \cdot L \\
&\quad L \rightarrow \cdot E , L \\
&\quad L \rightarrow \cdot E \\
&\quad E \rightarrow \cdot E \text{ and } E \\
&\quad E \rightarrow \cdot \text{mo} (L) \\
&\quad E \rightarrow \cdot \text{true} \\
&\quad E \rightarrow \cdot \text{false} \} \\
\text{GOTO}(I_9, \text{and}) &= I_5 \\
I_{12} &= \text{GOTO}(I_{11}, L) = \{ L \rightarrow E , L \cdot \} \\
\text{GOTO}(I_{11}, E) &= I_9 \\
\text{GOTO}(I_{11}, \text{mo}) &= I_2 \\
\text{GOTO}(I_{11}, \text{true}) &= I_3 \\
\text{GOTO}(I_{11}, \text{false}) &= I_4
\end{aligned}$$

A partir de la colección anterior, se construye la tabla de análisis SLR es la siguiente:

ESTADO	ACCIÓN							IR-A	
	and	mo	true	false	,	()	\$	E L
0		d2	d3	d4					1
1	d5							acc	
2									
3	r3				r3		r3	r3	
4	r4				r4		r4	r4	
5		d2	d3	d4					7
6		d2	d3	d4					9 8
7	d5/r1				r1		r1	r1	
8							d10		
9	d5				d11		r6		
10	r2				r2		r2	r2	
11		d2	d3	d4					9 12
12							r5		

La gramática no es SLR(1), ya que la tabla tiene un conflicto desplaza/reduce en el estado 7. El conflicto era previsible, ya que la gramática es ambigua. Por esta misma razón, no puede ser tampoco LALR(1) ni LR(1).

De los ítems del conjunto I_7 se deduce la acción que hay que seleccionar para que el operador **and** sea asociativo por la izquierda. En la respuesta del apartado 1 se muestra, a la derecha, el árbol de derivación que corresponde a la asociatividad por la izquierda de **and**. En la forma sentencial $E \text{ and } E \text{ and true}$ de la derivación correspondiente a este árbol se produce el conflicto, y se resuelve haciendo la reducción de $E \text{ and } E$ en lugar del desplazamiento.

Apartado 4.

La simulación de la cadena **true and false true** con tratamiento de errores en modo pánico es la siguiente:

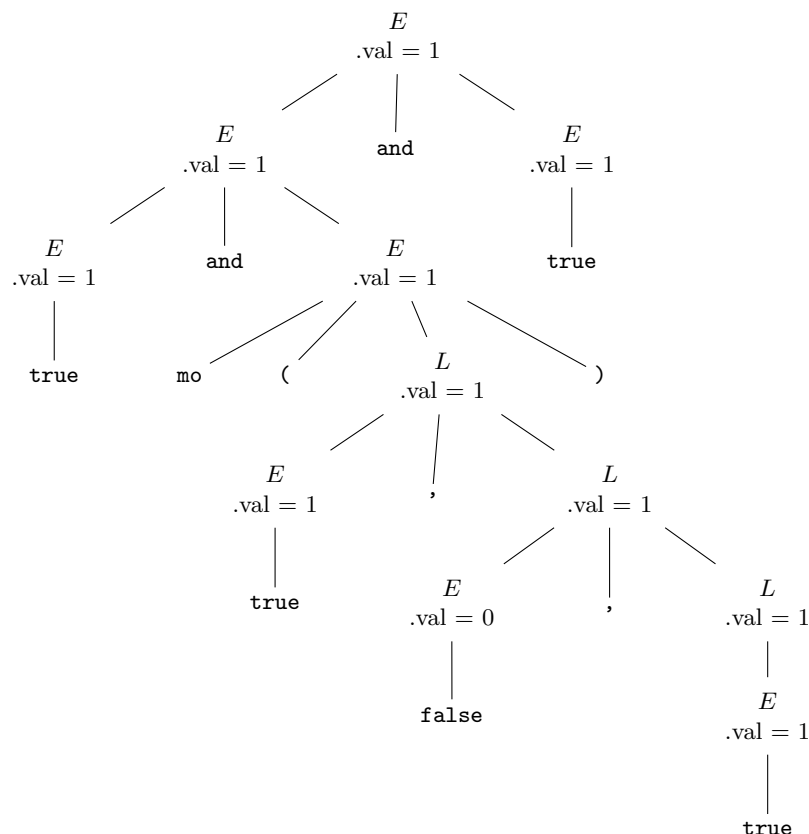
PILA	ENTRADA	ACCIÓN
0	true and false true \$	$d3$
0 true 3	and false true \$	$r3 E \rightarrow \text{true}$
0 E 1	and false true \$	$d5$
0 E 1 and 5	false true \$	$d4$
0 E 1 and 5 false 4	true \$	Error: desapila hasta estado 5; apila E y IR-A(5, E)
0 E 1 and 5 E 7	true \$	Eliminar tokens de la entrada hasta $\$ \in \text{SIG}(E)$
0 E 1 and 5 E 7	\$	$r1 E \rightarrow E \text{ and } E$
0 E 1	\$	Fin: entrada con errores

Apartado 5.

a) Para realizar la definición dirigida por la sintaxis de la gramática, sólo es necesario un atributo asociado a los no terminales E y L . Podemos denominar al atributo **val** y hacer que tome un valor 0 en caso de que el valor lógico asociado al no terminal sea falso, y 1 en caso de que sea verdadero. Empleando los operadores de C, podemos definir del siguiente modo las reglas semánticas:

Regla de producción	Regla semántica
$E \rightarrow E_1 \text{ and } E_2$	$E.\text{val} = E_1.\text{val} \ \&\& \ E_2.\text{val};$
$E \rightarrow \text{mo } (L)$	$E.\text{val} = L.\text{val};$
$E \rightarrow \text{true}$	$E.\text{val} = 1;$
$E \rightarrow \text{false}$	$E.\text{val} = 0;$
$L \rightarrow E , L_1$	$L.\text{val} = E.\text{val} \ \ L_1.\text{val};$
$L \rightarrow E$	$L.\text{val} = E.\text{val};$

b) El árbol anotado para la entrada **true and mo (true , false , true) and true** es el siguiente:



c) La DDS anterior define una gramática S-atribuida, porque los atributos que se emplean son todos sintetizados. Y por definición, toda gramática S-atribuida también es L-atribuida.