



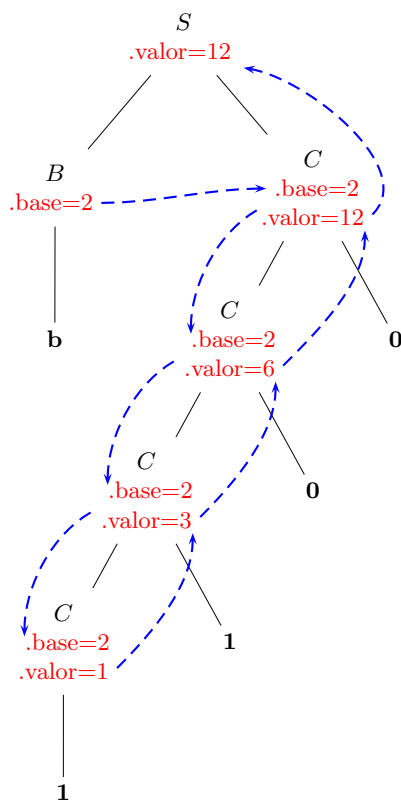
SOLUCIONES

Parte I: PREGUNTAS TIPO TEST. 30%.

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

Parte II: PREGUNTAS CORTAS. 10%.

Apartado 1.



Apartado 2.

- Conflicto [5,*]. Los ítems implicados son:

- $[R \rightarrow RR \cdot , *]$ (r2)
- $[R \rightarrow R \cdot *, a/b/|/ * / \$]$ (d6)

Este conflicto está relacionado con la precedencia de la operación de concatenación frente a la de clausura. Como esta última tiene más precedencia, debemos escoger la acción d6, para introducir en la pila el símbolo * y reducir posteriormente con $R \rightarrow R*$.

- Conflicto [5,a]. Los ítems implicados son:

- $[R \rightarrow RR \cdot , a]$ (r2)
- $[R \rightarrow \cdot a, a/b/|/ * / \$]$ (d2)

Este conflicto está relacionado con la asociatividad de la operación de concatenación. La asociatividad es por la izquierda, de modo que debemos escoger la acción r2. En caso contrario estaríamos forzando la asociatividad por la derecha.

- Conflicto [5,|]. Los ítems implicados son:

- $[R \rightarrow RR \cdot , |]$ (r2)
- $[R \rightarrow R \cdot | R, a/b/|/ * / \$]$ (d4)

Este conflicto está relacionado con la precedencia de la operación de unión frente a la de concatenación. Como esta última tiene más precedencia, debemos escoger la acción r2.

Parte III: PROBLEMA. 60%

Apartado 1.

G no es LL(1) porque no está factorizada. La parte derecha de las reglas del no terminal P tienen el prefijo común S , mientras que las reglas del no terminal S tienen el prefijo común E . La factorización introduce dos nuevos no terminales en la gramática, P' y S' , que eliminan esos prefijos comunes:

- 1) $P \rightarrow SP'$
- 2) $P' \rightarrow \&P$
- 3) $P' \rightarrow \lambda$
- 4) $S \rightarrow ES'$
- 5) $S' \rightarrow S$
- 6) $S' \rightarrow \lambda$
- 7) $E \rightarrow < I; O >$
- 8) $E \rightarrow (P)$
- 9) $I \rightarrow \mathbf{int}$
- 10) $O \rightarrow \mathbf{int}$

Esta nueva gramática es equivalente a la de partida, y puede ser LL(1).

Apartado 2.

Para comprobar si la gramática es LL(1) debemos calcular los conjuntos PRIMERO y SIGUIENTE:

$\text{PRIMERO}(P) = \{<, (\}$	$\text{SIGUIENTE}(P) = \{\$, \}$
$\text{PRIMERO}(P') = \{\&, \lambda\}$	$\text{SIGUIENTE}(P') = \{\$, \}$
$\text{PRIMERO}(S) = \{<, (\}$	$\text{SIGUIENTE}(S) = \{\&, \$, \}$
$\text{PRIMERO}(S') = \{<, (, \lambda\}$	$\text{SIGUIENTE}(S') = \{\&, \$, \}$
$\text{PRIMERO}(E) = \{<, (\}$	$\text{SIGUIENTE}(E) = \{<, (, \&, \$, \}$
$\text{PRIMERO}(I) = \{\mathbf{int}\}$	$\text{SIGUIENTE}(I) = \{;\}$
$\text{PRIMERO}(O) = \{\mathbf{int}\}$	$\text{SIGUIENTE}(O) = \{>\}$

Seguidamente, calculamos los conjuntos PREDICT de cada regla de producción:

$\text{PREDICT}(1) = \{<, (\}$	$\text{PREDICT}(6) = \{\&, \$, \}$
$\text{PREDICT}(2) = \{\&\}$	$\text{PREDICT}(7) = \{<\}$
$\text{PREDICT}(3) = \{\$, \}$	$\text{PREDICT}(8) = \{(\}$
$\text{PREDICT}(4) = \{<, (\}$	$\text{PREDICT}(9) = \{\mathbf{int}\}$
$\text{PREDICT}(5) = \{<, (\}$	$\text{PREDICT}(10) = \{\mathbf{int}\}$

La tabla de análisis LL(1) es la siguiente:

NO TERMINAL	SÍMBOLO DE ENTRADA							
	<	>	;	()	&	\$	int
P	1			1				
P'					3	2	3	
S	4			4				
S'	5			5	6	6	6	
E	7			8				
I								9
O								10

Puesto que la tabla no tiene ningún conflicto, podemos deducir que la gramática es LL(1).

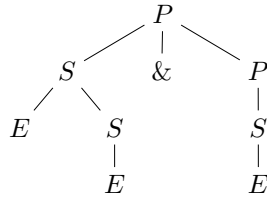
Apartado 3.

A continuación se muestra la simulación de la cadena de entrada $\langle \text{int}; \rangle$ con tratamiento de errores en modo pánico:

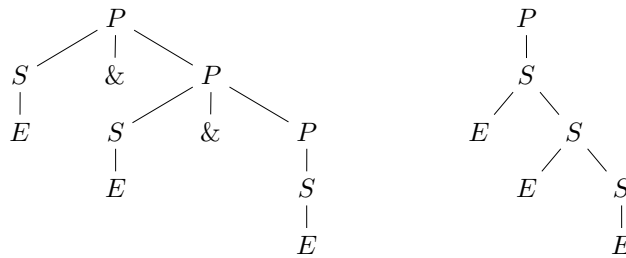
PILA	ENTRADA	SALIDA
\$ P	$\langle \text{int}; \rangle$	\$
\$ $P'S$	$\langle \text{int}; \rangle$	\$ $P \rightarrow SP'$
\$ $P'S'E$	$\langle \text{int}; \rangle$	\$ $S \rightarrow ES'$
\$ $P'S' > O; I <$	$\langle \text{int}; \rangle$	\$ $E \rightarrow \langle I; O \rangle$
\$ $P'S' > O; I$	$\text{int}; \rangle$	\$
\$ $P'S' > O; \text{int}$	$\text{int}; \rangle$	\$ $I \rightarrow \text{int}$
\$ $P'S' > O;$	$; \rangle$	\$
\$ $P'S' > O$	$>$	\$ Error: falta int . Activado modo pánico. \$ $> \in \text{SIGUIENTE}(O)$: desapilar O y continuar.
\$ $P'S' >$	$>$	\$
\$ $P'S'$		\$
\$ P'		\$ $S' \rightarrow \lambda$
\$		\$ $P' \rightarrow \lambda$
\$		\$ Aceptar

Apartado 4.

La propia gramática elimina la ambigüedad debida a la precedencia y asociatividad de las operaciones de agrupación serie y paralelo. El terminal P representa a componentes agrupados en paralelo, mientras que el terminal S representa a componentes agrupados en serie. Como S está en el lado derecho de las reglas de P , y no sucede lo contrario, la gramática establece una jerarquía en la que la conexión serie tiene más prioridad puesto que se reduce antes:



Por otra parte, debido al uso de recursividad por la derecha en las reglas de P y S , la asociatividad de la agrupación serie y paralelo es por la derecha también:



Apartado 5.

La lista completa de conjuntos de ítems es la siguiente:

$$I_0 = \{ \begin{array}{l} P' \rightarrow \cdot P \\ P \rightarrow \cdot S \\ P \rightarrow \cdot S \& P \\ S \rightarrow \cdot E \\ S \rightarrow \cdot ES \\ E \rightarrow \cdot \langle \text{int}; \text{int} \rangle \\ E \rightarrow \cdot (P) \end{array} \}$$

$$I_1 = \text{GOTO}(I_0, P) = \{ P' \rightarrow P \cdot \}$$

$$I_2 = \text{GOTO}(I_0, S) = \{ \begin{array}{l} P \rightarrow S \cdot \\ P \rightarrow S \cdot \& P \end{array} \}$$

$$I_3 = \text{GOTO}(I_0, E) = \{ \begin{array}{l} S \rightarrow E \cdot \\ S \rightarrow E \cdot S \\ S \rightarrow \cdot E \\ S \rightarrow \cdot ES \\ E \rightarrow \cdot \langle \text{int}; \text{int} \rangle \\ E \rightarrow \cdot (P) \end{array} \}$$

$$I_4 = \text{GOTO}(I_0, \langle \rangle) = \{ E \rightarrow \langle \cdot \text{int}; \text{int} \rangle \}$$

$$\begin{aligned}
I_5 = \text{GOTO}(I_0, ()) = \{ & E \rightarrow (\cdot P) \\
& P \rightarrow \cdot S \\
& P \rightarrow \cdot S \& P \\
& S \rightarrow \cdot E \\
& S \rightarrow \cdot ES \\
& E \rightarrow \cdot < \text{int}; \text{int} > \\
& E \rightarrow \cdot (P) \}
\end{aligned}$$

$$\begin{aligned}
I_6 = \text{GOTO}(I_2, \&) = \{ & P \rightarrow S \& \cdot P \\
& P \rightarrow \cdot S \\
& P \rightarrow \cdot S \& P \\
& S \rightarrow \cdot E \\
& S \rightarrow \cdot ES \\
& E \rightarrow \cdot < \text{int}; \text{int} > \\
& E \rightarrow \cdot (P) \}
\end{aligned}$$

$$I_7 = \text{GOTO}(I_3, S) = \{ S \rightarrow ES \cdot \}$$

$$\text{GOTO}(I_3, E) = I_3$$

$$\text{GOTO}(I_3, <) = I_4$$

$$\text{GOTO}(I_3, ()) = I_5$$

$$I_8 = \text{GOTO}(I_4, \text{int}) = \{ E \rightarrow < \text{int}; \text{int} > \cdot \}$$

$$I_9 = \text{GOTO}(I_5, P) = \{ E \rightarrow (P \cdot) \}$$

$$\text{GOTO}(I_5, S) = I_2$$

$$\text{GOTO}(I_5, E) = I_3$$

$$\text{GOTO}(I_5, <) = I_4$$

$$\text{GOTO}(I_5, ()) = I_5$$

$$I_{10} = \text{GOTO}(I_6, P) = \{ P \rightarrow S \& P \cdot \}$$

$$\text{GOTO}(I_6, S) = I_2$$

$$\text{GOTO}(I_6, E) = I_3$$

$$\text{GOTO}(I_6, <) = I_4$$

$$\text{GOTO}(I_6, ()) = I_5$$

$$I_{11} = \text{GOTO}(I_8, ;) = \{ E \rightarrow < \text{int}; \cdot \text{int} > \}$$

$$I_{12} = \text{GOTO}(I_9,) = \{ E \rightarrow (P) \cdot \}$$

$$I_{13} = \text{GOTO}(I_{11}, \text{int}) = \{ E \rightarrow < \text{int}; \text{int} \cdot > \}$$

$$I_{14} = \text{GOTO}(I_{13}, >) = \{ E \rightarrow < \text{int}; \text{int} > \cdot \}$$

Para construir la tabla de análisis es necesario calcular los conjuntos PRIMERO y SIGUIENTE de la gramática de este apartado:

$$\begin{aligned}
\text{PRIMERO}(P) &= \{ <, (\} \\
\text{PRIMERO}(S) &= \{ <, (\} \\
\text{PRIMERO}(E) &= \{ <, (\}
\end{aligned}$$

$$\begin{aligned}
\text{SIGUIENTE}(P) &= \{ \$,) \} \\
\text{SIGUIENTE}(S) &= \{ \&, \$,) \} \\
\text{SIGUIENTE}(E) &= \{ <, (, \&, \$,) \}
\end{aligned}$$

La tabla de análisis es la siguiente:

ESTADO	ACCIÓN								IR-A		
	&	()	;	<	>	int	\$	P	S	E
0		d5			d4				1	2	3
1								acc			
2	d6		r1					r1			
3	r3	d5	r3		d4			r3		7	3
4							d8				
5		d5			d4				9	2	3
6		d5			d4				10	2	3
7	r4		r4					r4			
8				d11							
9			d12								
10			r2					r2			
11							d13				
12	r6	r6	r6		r6			r6			
13						d14					
14	r5	r5	r5		r5			r5			

La gramática es SLR, ya que la tabla no contiene ningún conflicto.

Apartado 6.

Emplearemos los siguientes atributos:

Símbolo	Atributo	Tipo	Comentario
P	in	int	Número de interfaces de entrada de P .
	out	int	Número de interfaces de salida de P .
S	in	int	Número de interfaces de entrada de S .
	out	int	Número de interfaces de salida de S .
E	in	int	Número de interfaces de entrada de E .
	out	int	Número de interfaces de salida de E .
int	val	int	Valor numérico del lexema entero.

La definición dirigida por la sintaxis (DDS) para verificar la corrección semántica de las agrupaciones de componentes es la siguiente:

Regla de producción	Acción
$P \rightarrow S$	$P.in = S.in; P.out = S.out;$
$P \rightarrow S \& P_1$	$P.in = S.in + P_1.in; P.out = S.out + P_1.out;$
$S \rightarrow E$	$S.in = E.in; S.out = E.out;$
$S \rightarrow ES_1$	if ($E.out \neq S_1.in$) error(); else $S.in = E.in; S.out = S_1.out;$
$E \rightarrow \langle \text{int}_1; \text{int}_2 \rangle$	$E.in = \text{int}_1.in; E.out = \text{int}_2.out;$
$E \rightarrow (P)$	$E.in = P.in; E.out = P.out;$

En la DDS sólo hay atributos sintetizados, de modo que es S-atribuida y L-atribuida.

Para la entrada $\langle 1; 2 \rangle \langle 2; 1 \rangle \& \langle 3; 2 \rangle$, el árbol anotado es el siguiente:

