



SOLUCIONES

Parte I: PREGUNTAS TIPO TEST. 30%.

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

Parte II: PROBLEMA. 70%.

Apartado 1.

$\text{PRIMERO}(S) = \{ id \}$
 $\text{PRIMERO}(T) = \{ \neg, (, id \}$

$\text{SIGUIENTE}(S) = \{ \$ \}$
 $\text{SIGUIENTE}(T) = \{ \vee, \Rightarrow,), \$ \}$

La gramática no es LL(1) porque es ambigua (reglas $T \rightarrow T \vee T \mid T \Rightarrow T$), es recursiva por la izquierda y tiene factores comunes (en las mismas reglas de T).

Apartado 2

Para eliminar el problema de la ambigüedad, usamos una solución similar a la empleada en la gramática no ambigua de las expresiones aritméticas, teniendo en cuenta la asociatividad izquierda de los operadores y la precedencia:

$$\begin{aligned}
 S &\rightarrow id = E \\
 E &\rightarrow E \Rightarrow T \mid T \\
 T &\rightarrow T \vee F \mid F \\
 F &\rightarrow \neg F \mid (E) \mid id
 \end{aligned}$$

La gramática no está todavía preparada para el análisis LL(1) puesto que tiene recursión inmediata por la izquierda en E y T . Eliminamos la recursión inmediata:

$$\begin{aligned}
 S &\rightarrow id = E \\
 E &\rightarrow T \mid TE' \\
 E' &\rightarrow \Rightarrow T \mid \Rightarrow TE' \\
 T &\rightarrow F \mid FT' \\
 T' &\rightarrow \vee F \mid \vee FT' \\
 F &\rightarrow \neg F \mid (E) \mid id
 \end{aligned}$$

Se debe dar un último paso antes de poder intentar el análisis LL(1): factorizar las reglas de producción:

$$\begin{aligned}
 S &\rightarrow id = E \\
 E &\rightarrow TE' \\
 E' &\rightarrow \lambda \mid \Rightarrow TE' \\
 T &\rightarrow FT' \\
 T' &\rightarrow \lambda \mid \vee FT' \\
 F &\rightarrow \neg F \mid (E) \mid id
 \end{aligned}$$

Los conjuntos PRIMERO y SIGUIENTE de los no terminales de la gramática modificada son:

$\text{PRIMERO}(S) = \{ id \}$
 $\text{PRIMERO}(E) = \{ \neg (id) \}$
 $\text{PRIMERO}(E') = \{ \lambda \Rightarrow \}$
 $\text{PRIMERO}(T) = \{ \neg (id) \}$
 $\text{PRIMERO}(T') = \{ \lambda \vee \}$
 $\text{PRIMERO}(F) = \{ \neg (id) \}$

$\text{SIGUIENTE}(S) = \{ \$ \}$
 $\text{SIGUIENTE}(E) = \{) \$ \}$
 $\text{SIGUIENTE}(E') = \{) \$ \}$
 $\text{SIGUIENTE}(T) = \{ \Rightarrow) \$ \}$
 $\text{SIGUIENTE}(T') = \{ \Rightarrow) \$ \}$
 $\text{SIGUIENTE}(F) = \{ \vee \Rightarrow) \$ \}$

Los conjuntos PREDICT de las reglas de producción de la gramática modificada son:

$\text{PREDICT}(S \rightarrow id = E) = \{ id \}$
 $\text{PREDICT}(E \rightarrow TE') = \{ \neg (id) \}$
 $\text{PREDICT}(E' \rightarrow \lambda) = \{) \$ \}$
 $\text{PREDICT}(E' \rightarrow \Rightarrow TE') = \{ \Rightarrow \}$
 $\text{PREDICT}(T \rightarrow FT') = \{ \neg (id) \}$

$\text{PREDICT}(T' \rightarrow \lambda) = \{ \Rightarrow) \$ \}$
 $\text{PREDICT}(T' \rightarrow \vee FT') = \{ \vee \}$
 $\text{PREDICT}(F \rightarrow \neg F) = \{ \neg \}$
 $\text{PREDICT}(F \rightarrow (E)) = \{ (\}$
 $\text{PREDICT}(F \rightarrow id) = \{ id \}$

De acuerdo con los conjuntos PREDICT anteriores, **la gramática modificada es LL(1)** puesto que la intersección de los conjuntos PREDICT de las reglas de un mismo no terminal es \emptyset .

Apartado 3.

Obviamente, la gramática no es SLR(1), LR(1) ni LALR(1), puesto que es ambigua.

Calculamos la colección LR(0):

$$I_0 = \{ S' \rightarrow \cdot S \\ S \rightarrow \cdot id = T \}$$

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

$$I_2 = \text{GOTO}(I_0, id) = \{ S \rightarrow id \cdot = T \}$$

$$I_3 = \text{GOTO}(I_2, =) = \{ S \rightarrow id = \cdot T \\ T \rightarrow \cdot T \vee T \\ T \rightarrow \cdot T \Rightarrow T \\ T \rightarrow \cdot \neg T \\ T \rightarrow \cdot (T) \\ T \rightarrow \cdot id \}$$

$$I_4 = \text{GOTO}(I_3, T) = \{ S \rightarrow id = T \cdot \\ T \rightarrow T \cdot \vee T \\ T \rightarrow T \cdot \Rightarrow T \}$$

$$I_5 = \text{GOTO}(I_3, \neg) = \{ T \rightarrow \neg \cdot T \\ T \rightarrow \cdot T \vee T \\ T \rightarrow \cdot T \Rightarrow T \\ T \rightarrow \cdot \neg T \\ T \rightarrow \cdot (T) \\ T \rightarrow \cdot id \}$$

$$I_6 = \text{GOTO}(I_3, () = \{ T \rightarrow (\cdot T) \\ T \rightarrow \cdot T \vee T \\ T \rightarrow \cdot T \Rightarrow T \\ T \rightarrow \cdot \neg T \\ T \rightarrow \cdot (T) \\ T \rightarrow \cdot id \}$$

$$I_7 = \text{GOTO}(I_3, id) = \{ T \rightarrow id \cdot \}$$

$$I_8 = \text{GOTO}(I_4, \vee) = \{ T \rightarrow T \vee \cdot T) \\ T \rightarrow \cdot T \vee T \\ T \rightarrow \cdot T \Rightarrow T \\ T \rightarrow \cdot \neg T \\ T \rightarrow \cdot (T) \\ T \rightarrow \cdot id \}$$

$$I_9 = \text{GOTO}(I_4, \Rightarrow) = \{ T \rightarrow T \Rightarrow \cdot T) \\ T \rightarrow \cdot T \vee T \\ T \rightarrow \cdot T \Rightarrow T \\ T \rightarrow \cdot \neg T \\ T \rightarrow \cdot (T) \\ T \rightarrow \cdot id \}$$

$$I_{10} = \text{GOTO}(I_5, T) = \{ T \rightarrow \neg T \cdot \\ T \rightarrow T \cdot \vee T \\ T \rightarrow T \cdot \Rightarrow T \}$$

$$\text{GOTO}(I_5, \neg) = I_5 \\ \text{GOTO}(I_5, () = I_6 \\ \text{GOTO}(I_5, id) = I_7$$

$$I_{11} = \text{GOTO}(I_6, T) = \{ T \rightarrow (T \cdot) \\ T \rightarrow T \cdot \vee T \\ T \rightarrow T \cdot \Rightarrow T \}$$

$$\text{GOTO}(I_6, \neg) = I_5 \\ \text{GOTO}(I_6, () = I_6 \\ \text{GOTO}(I_6, id) = I_7$$

$$I_{12} = \text{GOTO}(I_8, T) = \{ \begin{array}{l} T \rightarrow T \vee T \cdot \\ T \rightarrow T \cdot \vee T \\ T \rightarrow T \cdot \Rightarrow T \cdot \end{array} \}$$

$$\begin{aligned} \text{GOTO}(I_8, \neg) &= I_5 \\ \text{GOTO}(I_8, () &= I_6 \\ \text{GOTO}(I_8, id) &= I_7 \end{aligned}$$

$$I_{13} = \text{GOTO}(I_9, T) = \{ \begin{array}{l} T \rightarrow T \Rightarrow T \cdot \\ T \rightarrow T \cdot \vee T \\ T \rightarrow T \cdot \Rightarrow T \cdot \end{array} \}$$

$$\begin{aligned} \text{GOTO}(I_9, \neg) &= I_5 \\ \text{GOTO}(I_9, () &= I_6 \\ \text{GOTO}(I_9, id) &= I_7 \end{aligned}$$

$$\begin{aligned} \text{GOTO}(I_{10}, \vee) &= I_8 \\ \text{GOTO}(I_{10}, \Rightarrow) &= I_9 \end{aligned}$$

$$I_{14} = \text{GOTO}(I_{11},) = \{ T \rightarrow (T) \cdot \}$$

$$\begin{aligned} \text{GOTO}(I_{11}, \vee) &= I_8 \\ \text{GOTO}(I_{11}, \Rightarrow) &= I_9 \end{aligned}$$

$$\begin{aligned} \text{GOTO}(I_{12}, \vee) &= I_8 \\ \text{GOTO}(I_{12}, \Rightarrow) &= I_9 \end{aligned}$$

$$\begin{aligned} \text{GOTO}(I_{13}, \vee) &= I_8 \\ \text{GOTO}(I_{13}, \Rightarrow) &= I_9 \end{aligned}$$

Para determinar las acciones de reducción, es necesario obtener previamente los conjuntos SIGUIENTE de los no terminales de la gramática inicial:

$$\begin{aligned} \text{PRIMERO}(S) &= \{ id \} \\ \text{PRIMERO}(T) &= \{ \neg (id \} \end{aligned}$$

$$\begin{aligned} \text{SIGUIENTE}(S) &= \{ \$ \} \\ \text{SIGUIENTE}(T) &= \{ \vee \Rightarrow) \$ \} \end{aligned}$$

A partir de los cálculos anteriores, podemos construir la siguiente tabla de análisis SLR:

ESTADO	ACCIÓN							IR-A	
	<i>id</i>	=	\vee	\Rightarrow	\neg	()	\$	<i>S</i> <i>T</i>
0	d2								1
1								acc	
2		d3							
3	d7				d5	d6			4
4			d8	d9				r1	
5	d7				d5	d6			10
6	d7				d5	d6			11
7			r6	r6				r6	r6
8	d7				d5	d6			12
9	d7				d5	d6			13
10			d8/r4	d9/r4				r4	r4
11			d8	d9				d14	
12			d8/r2	d9/r2				r2	r2
13			d8/r3	d9/r3				r3	r3
14			r5	r5				r5	r5

La numeración de las reglas de producción usada en la tabla es la siguiente:

1. $S \rightarrow id = T$
2. $T \rightarrow T \vee T$
3. $T \rightarrow T \Rightarrow T$
4. $T \rightarrow \neg T$
5. $T \rightarrow (T)$
6. $T \rightarrow id$

Los conflictos de la tabla se resuelven usando las precedencias y asociatividades de los operadores lógicos. La mayor precedencia corresponde a \neg , luego a \vee y por último a \Rightarrow . Estos dos últimos operadores tienen asociatividad izquierda.

- Los conflictos $T[10, \vee]$ y $T[10, \Rightarrow]$ se resuelven empleando r4, por mayor precedencia de \neg .
- El conflicto $T[12, \vee]$ se resuelve empleando r2, por asociatividad izquierda de \vee .

- El conflicto $T[12, \Rightarrow]$ se resuelve empleando $r2$, por mayor precedencia de \vee frente a \Rightarrow .
- El conflicto $T[13, \vee]$ se resuelve empleando $d8$, por mayor precedencia de \vee frente a \Rightarrow .
- El conflicto $T[13, \Rightarrow]$ se resuelve empleando $r3$, por asociatividad izquierda de \Rightarrow .

Apartado 4.

La simulación de la cadena $id = id \Rightarrow \vee id$ con tratamiento de errores en modo pánico es la siguiente:

PILA	ENTRADA	ACCIÓN
0	$id = id \Rightarrow \vee id \$$	$d2$
0 id 2	$= id \Rightarrow \vee id \$$	$d3$
0 id 2 = 3	$id \Rightarrow \vee id \$$	$d7$
0 id 2 = 3 id 7	$\Rightarrow \vee id \$$	$r6 \quad T \rightarrow id$
0 id 2 = 3 T 4	$\Rightarrow \vee id \$$	$d9$
0 id 2 = 3 T 4 \Rightarrow 9	$\vee id \$$	Error: usar IR-A[9,T] y $\vee \in \text{SIGUIENTE}(T)$
0 id 2 = 3 T 4 \Rightarrow 9 T 13	$\vee id \$$	$d8$
0 id 2 = 3 T 4 \Rightarrow 9 T 13 \vee 8	$id \$$	$d7$
0 id 2 = 3 T 4 \Rightarrow 9 T 13 \vee 8 id 7	$\$$	$r6 \quad T \rightarrow id$
0 id 2 = 3 T 4 \Rightarrow 9 T 13 \vee 8 T 12	$\$$	$r2 \quad T \rightarrow T \vee T$
0 id 2 = 3 T 4 \Rightarrow 9 T 13	$\$$	$r3 \quad T \rightarrow T \Rightarrow T$
0 id 2 = 3 T 4	$\$$	$r1 \quad S \rightarrow id = T$
0 S 1	$\$$	No acepta por errores

Apartado 5.

Para realizar la definición dirigida por la sintaxis que se solicita, emplearemos los siguientes atributos:

Símbolo	Atributo	Descripción
id	lex	Lexema del identificador
T	equiv	Cadena de texto con la fórmula equivalente sin el operador \Rightarrow
T	neg	Cadena de texto con la fórmula equivalente sin el operador \Rightarrow y negada
S	equiv	Cadena de texto con la fórmula equivalente sin el operador \Rightarrow

Usando el operador $+$ como concatenador de cadenas de texto, las reglas semánticas para la evaluación de los atributos de cada símbolo son:

Regla de producción	Regla semántica
$T \rightarrow id$	$T.\text{equiv} = id.\text{lex}; T.\text{neg} = "\neg" + id.\text{lex};$
$T \rightarrow (T_1)$	$T.\text{equiv} = "(" + T_1.\text{equiv} + ")"; T.\text{neg} = "(" + T_1.\text{neg} + ")";$
$T \rightarrow \neg T_1$	$T.\text{equiv} = T_1.\text{neg}; T.\text{neg} = T_1.\text{equiv};$
$T \rightarrow T_1 \Rightarrow T_2$	$T.\text{equiv} = T_1.\text{neg} + "\vee" + T_2.\text{equiv}; T.\text{neg} = T_1.\text{equiv} + "\wedge" + T_2.\text{neg};$
$T \rightarrow T_1 \vee T_2$	$T.\text{equiv} = T_1.\text{equiv} + "\vee" + T_2.\text{equiv}; T.\text{neg} = T_1.\text{neg} + "\wedge" + T_2.\text{neg};$
$S \rightarrow id = T$	$S.\text{equiv} = T.\text{equiv};$

Por el modo de evaluar los atributos, todos ellos son sintetizados. Por tanto, la gramática atribuida es S-atribuida. También es L-atribuida, porque por definición las gramáticas L-atribuidas incluyen a las S-atribuidas.

Con esta DDS, para la cadena solicitada en el enunciado, podemos construir un árbol sintáctico decorado como el que se indica en la siguiente página.

