

6. 1.0 puntos Dada la siguiente gramática:

```

programa → funciones main
          | main
funciones → funciones funcion
          | funcion
funcion   → 'function' ident '(' bloque
main      → 'main' bloque
bloque    → '{' sentencias '}'
sentencias → sentencias sentencia
          | sentencia
sentencia → ident '=' expresion ';'
          | bloque
expresion → expresion '[' ident ']' ident
          | expresion '?' ident ':' ident
          | expresion '+' const
          | '*' expresion
          | '&' expresion
          | ident
          | '[' ident ']'
          | const
ident     → ident letra
          | letra
letra     → 'a' | 'b' | ... | 'z'
const     → num '.' num
num       → '0' | '1' | ... | '9'

```

Obtener la tabla de tokens con el máximo nivel de abstracción suponiendo que se va a realizar:

(a) 0.7 puntos Traducción.

Solución:

La tabla de tokens con el máximo nivel de abstracción cuando se realiza el proceso completo de traducción sería la siguiente:

Nº	Token	Patrón
1	CONST	<code>[0-9]+ "." [0-9]+</code>
2	IDENT	<code>"[a-z]+"</code>
3	CORI	<code>"["</code>
4	CORD	<code>"]"</code>
5	AMPAST	<code>"&" "*"</code>
6	MAS	<code>"+"</code>
7	INTER	<code>"?"</code>
8	DOSP	<code>":"</code>
9	ASIGN	<code>"="</code>
10	PYC	<code>";"</code>
11	LLAVEI	<code>"{"</code>
12	LLAVED	<code>"}"</code>
13	MAIN	<code>"main"</code>
14	FUNCTION	<code>"function"</code>
15	PARENT	<code>"("</code>

(b) 0.3 puntos Análisis sintáctico.

Solución:

Cuando únicamente se va a realizar análisis sintáctico, primero se observa que la gramática abstracta resultante del proceso anterior es libre de contexto debido a una producción en la forma de construcción de una sentencia:

sentencia → bloque

Por lo tanto, sería posible realizar una abstracción mayor hasta el símbolo no terminal `expresion`, cuyas formas de producción obedecen a una gramática regular. De esta forma, la tabla de tokens con máximo nivel de abstracción quedaría así:

Nº	Token	Patrón
1	SENTASIG	[a-b]+="{expression}"; "
2	LLAVEI	"{"
3	LLAVED	"}"
4	MAIN	"main"
5	CABFUNCTION	"function()"

Donde {expression} representa el patrón de construcción de las formas de producción de una expression.

7. 0.5 puntos Dada la gramática siguiente:

$$\begin{array}{lcl}
 A & \rightarrow & A \, b \, B \, Z \\
 & | & \epsilon \\
 B & \rightarrow & c \, d \\
 & | & B \, d \, b \\
 & | & \epsilon \\
 Z & \rightarrow & g \, Z \\
 & | & \epsilon
 \end{array}$$

Realizar los cambios que se estimen oportunos sobre la misma y construir la tabla de análisis LL(1).

Solución:

Se observan varias producciones con recursividad a la izquierda. Los símbolos no terminales que aparecen con recursividad a la izquierda son A y B . Tras eliminar la recursividad a la izquierda nos queda la siguiente gramática equivalente:

$$\begin{array}{lcl}
 A & \rightarrow & A' \\
 A' & \rightarrow & b \, B \, Z \, A' \\
 & | & \epsilon \\
 B & \rightarrow & c \, d \, B' \\
 & | & B' \\
 B' & \rightarrow & d \, b \, B' \\
 & | & \epsilon \\
 Z & \rightarrow & g \, Z \\
 & | & \epsilon
 \end{array}$$

No es necesario factorizar la gramática anterior. Para la construcción de la tabla de análisis LL(1) calculamos los iniciales y seguidores.

Iniciales(A) = {b, ϵ }	Seguidores(A) = {\$}
Iniciales(B) = {c, d, ϵ }	Seguidores(B) = {g, b, \$}
Iniciales(Z) = {g, ϵ }	Seguidores(Z) = {b, \$}
Iniciales(A') = {b, ϵ }	Seguidores(A') = {\$}
Iniciales(B') = {d, ϵ }	Seguidores(B') = {g, b, \$}

Por lo que la tabla de análisis LL(1) quedaría así:

	b	c	d	g	\$
A	$A \rightarrow A'$				$A \rightarrow A'$
B	$B \rightarrow B'$	$B \rightarrow cdB'$	$B \rightarrow B'$	$B \rightarrow B'$	$B \rightarrow B'$
Z	$Z \rightarrow \epsilon$			$Z \rightarrow gZ$	$Z \rightarrow \epsilon$
A'	$A' \rightarrow bBZA'$				$A' \rightarrow \epsilon$
B'	$B' \rightarrow \epsilon$		$B' \rightarrow dbB'$	$B' \rightarrow \epsilon$	$B' \rightarrow \epsilon$

No presenta conflictos la tabla de análisis, por lo tanto, se trata de una gramática LL(1).

8. 0.2 puntos Dada la gramática del problema 7, completar las acciones que debe aparecer en el estado inicial I_0 de la tabla de análisis LR(1).

Estado	b	c	d	g	\$	A	B	Z
I_0								

Solución:

Los items del estado I_0 son los siguientes:

$$I_0 = \{[A' \rightarrow \cdot A, \$], [A \rightarrow \cdot AbBZ, \$ / b], [A \rightarrow \epsilon \cdot, \$ / b]\}$$

A partir de dicho estado la única transición constituiría el siguiente estado:

$$I_1 = goto(I_0, A) = \{[A' \rightarrow A \cdot, \$], [A \rightarrow A \cdot bBZ, \$ / b]\}$$

Por lo tanto, el estado I_0 de la tabla de análisis LR(1) quedaría así:

Estado	b	c	d	g	\$	A	B	Z
I_0	r_2				r_2	1		