

1. Considera la siguiente gramática:

$$P = \{ \begin{array}{l} \langle \text{oración} \rangle \rightarrow \langle \text{sujeto} \rangle \langle \text{predicado} \rangle \\ \langle \text{sujeto} \rangle \rightarrow \langle \text{grupo_nominal} \rangle \\ \langle \text{sujeto} \rangle \rightarrow \langle \text{grupo_nominal} \rangle \text{ADJETIVO} \\ \langle \text{grupo_nominal} \rangle \text{ NOMBRE} \mid \text{ARTÍCULO NOMBRE} \\ \langle \text{predicado} \rangle \rightarrow \text{VERBO} \langle \text{complementos} \rangle \\ \langle \text{complementos} \rangle \rightarrow \langle \text{directo} \rangle \langle \text{indirecto} \rangle \langle \text{circunstanciales} \rangle \\ \langle \text{complementos} \rangle \rightarrow \langle \text{indirecto} \rangle \langle \text{circunstanciales} \rangle \\ \langle \text{complementos} \rangle \rightarrow \langle \text{directo} \rangle \langle \text{circunstanciales} \rangle \\ \langle \text{complementos} \rangle \rightarrow \langle \text{circunstanciales} \rangle \\ \langle \text{directo} \rangle \rightarrow \langle \text{grupo_nominal} \rangle \mid \langle \text{grupo_nominal} \rangle \text{ADJETIVO} \\ \langle \text{indirecto} \rangle \rightarrow \text{PREPOSICIÓN} \langle \text{grupo_nominal} \rangle \\ \langle \text{circunstanciales} \rangle \rightarrow \epsilon \\ \langle \text{circunstanciales} \rangle \rightarrow \langle \text{circunstancial} \rangle \langle \text{circunstanciales} \rangle \\ \langle \text{circunstancial} \rangle \rightarrow \text{PREPOSICIÓN} \langle \text{grupo_nominal} \rangle \end{array} \}$$

- Indica cuáles son los símbolos terminales y cuáles los no terminales
- ¿Cuál es el símbolo inicial?
- ¿Es la gramática recursiva por la izquierda o por la derecha?
- Muestra las derivaciones por la izquierda y por la derecha de la siguiente cadena
 - *Los Reyes Magos entregaron la bicicleta amarilla a la niña en el colegio*
- Dibuja el árbol sintáctico asociado a la derivación por la izquierda.

2. Considera la siguiente gramática:

$$P = \{ \begin{array}{l} S \rightarrow (A) \mid a \\ A \rightarrow a , S \mid b \end{array} \}$$

- Indica cuáles son los símbolos terminales y cuáles los no terminales
- ¿Cuál es el símbolo inicial?
- ¿Es la gramática recursiva por la izquierda o por la derecha?
- Muestra las derivaciones por la izquierda y por la derecha de la siguiente cadena
- (a , (a , (b)))
- Muestra los árboles sintácticos de las derivaciones del apartado anterior

3. Dada la siguiente gramática

$$P = \{ \begin{array}{l} A \rightarrow \text{IDENTIFICADOR} = E \\ E \rightarrow T \mid E + T \\ T \rightarrow T * P \mid P \\ P \rightarrow F \mid F \wedge P \\ F \rightarrow (E) \mid \text{NÚMERO} \mid \text{IDENTIFICADOR} \end{array} \}$$

- Elimina la recursividad por la izquierda y factorízala por la izquierda
- Utiliza la gramática obtenida en el apartado anterior para generar la derivación por la izquierda y el árbol sintáctico de la siguiente sentencia:
 - $h = a^2 + b^2$

4. Elimina la recursividad por la izquierda y factoriza por la izquierda las siguientes gramáticas

$$P1 = \{ \begin{array}{l} S \rightarrow S D ; \mid D ; \\ D \rightarrow T \text{ IDENTIFICADOR } (L) \\ T \rightarrow T * \mid \text{INT} \end{array} \}$$

$$L \rightarrow \varepsilon \mid I$$

$$I \rightarrow T \mid T, I\}$$

$$P2 = \{ \begin{array}{l} S \rightarrow S D ; \mid D ; \\ D \rightarrow T L \\ T \rightarrow \text{INT} \mid \text{FLOAT} \\ L \rightarrow I \mid I, L \\ I \rightarrow * I \mid \text{IDENTIFICADOR} \\ \end{array} \}$$

5. Dada la siguiente gramática:

$$P = \{ \begin{array}{l} P \rightarrow \text{inicio } S \text{ fin} \\ S \rightarrow D I \\ D \rightarrow D D' ; \mid D' ; \\ D' \rightarrow \text{IDENTIFICADOR } L' T \\ L' \rightarrow , \text{IDENTIFICADOR } L' \mid \varepsilon \\ T \rightarrow \text{REAL} \mid \text{ENTERO} \mid \text{CARÁCTER} \\ I \rightarrow I I' ; \mid I' ; \\ I' \rightarrow \text{ASIGNACIÓN} \mid \text{BUCLE} \\ \end{array} \}$$

- Elimina la recursividad por la izquierda y factorízala por la izquierda
- Utiliza la gramática obtenida en el apartado anterior para generar la derivación y el árbol sintáctico de las siguientes sentencias

inicio

dato entero;
a, b carácter;

bucle ;
asignación ;
bucle ;

fin

6. Diseña gramáticas de contexto libre que generen los lenguajes que se indican

- $L1 = \{ x \mid x \text{ es de la forma } a y b, \text{ donde } y \text{ es cualquier secuencia de ceros o unos} \}$
- $L2 = \{ a^i b^i \mid i > 0 \}$
- $L3 = \{ a^i b^{2i} \mid i > 0 \}$
- $L4 = \{ x \mid x \text{ tiene igual número de ceros que de unos} \}$
- $L5 = \{ x \mid x \text{ tiene diferente número de ceros que de unos} \}$
- $L6 = \{ w w R \mid w \in \{0,1\}^* \text{ y } w R \text{ es la palabra inversa o refleja de } w \}$
- $L7 = \{ w c w R \mid w \in \{a,b\}^* \}$

7. Diseña gramáticas de contexto libre que permitan genera las siguientes sentencias del lenguaje de programación C:

- Declaraciones de arrays de tipo int o float

Ejemplo:
int a[3][3], b[5];
float imagen[3][];

- Declaraciones de punteros a variables de tipo int o float

Ejemplo
*float **puntero_doble, *a;*

- Declaraciones de variables simples, arrays o punteros a variables o arrays de los tipos int, o float:

Ejemplo
`int a, b[2], *c[3][4];`
`int corto, *d;`
`float x, *y, **z[];`

- d. Proposiciones lógicas, como por ejemplo:
 $(a == b) \ \&\& \ (c != 10 \ || \ a >= 1)$
- e. Sentencias de control en C: if, while, do ... while, for, switch.
- f. Prototipos de funciones que devuelvan enteros o punteros a enteros y tengan cuyos parámetros sean del tipo int, float o puntero (simple, doble, ...) a int o float.

8. Diseña una gramática que permite generar expresiones aritméticas con notación prefija.

Ejemplo
 $(+ (* a a) (* b b))$
3
 $(- 3)$
 $(/ (- b 2) (- b 1))$

9. Dada la siguiente gramática

$$P = \{ S \rightarrow a \mid S a \mid b S S \mid S S b \mid S b S \}$$

- a. Comprueba que es ambigua generando dos derivaciones por la izquierda (o por la derecha) diferentes.
 - b. Construye los árboles sintácticos asociados a esas derivaciones.
10. Comprueba que la siguiente gramática es ambigua y diseña otra equivalente que no lo sea.

$$P = \{ \begin{array}{l} S \rightarrow A \mid B \\ A \rightarrow a A b \mid a b \\ B \rightarrow a b B \mid \epsilon \end{array} \}$$

11. Compruébese que la siguiente gramática es LL(1) sin modificarla.

$$\begin{array}{l} A \rightarrow B C D \\ B \rightarrow a C b \\ B \rightarrow \epsilon \\ C \rightarrow c A d \\ C \rightarrow e B f \\ C \rightarrow g D h \\ C \rightarrow \epsilon \\ D \rightarrow l \end{array}$$

12. ¿ Es LL(1) la siguiente gramática ?

$$\begin{array}{l} A \rightarrow B C D \\ B \rightarrow b \\ B \rightarrow \epsilon \\ C \rightarrow c \\ C \rightarrow \epsilon \\ D \rightarrow d \\ D \rightarrow \epsilon \end{array}$$

13. Dada la siguiente gramática, elimínese la recursividad a la izquierda y los factores comunes por la izquierda y compruébese que la gramática resultante cumple la condición LL(1).

$S \rightarrow S \text{ inst}$
 $S \rightarrow S \text{ var } D$
 $S \rightarrow \epsilon$
 $D \rightarrow D \text{ ident } E$
 $D \rightarrow D \text{ ident sep}$
 $D \rightarrow \text{int}$
 $D \rightarrow \text{float}$
 $E \rightarrow S \text{ fproc}$

14. Dada la siguiente gramática:

$S \rightarrow A B$
 $A \rightarrow \text{begin } S \text{ end } B \text{ theend}$
 $A \rightarrow \epsilon$
 $B \rightarrow \text{var } L : \text{tipo}$
 $B \rightarrow B \text{ fvar}$
 $B \rightarrow \epsilon$
 $L \rightarrow L , \text{id}$
 $L \rightarrow \text{id}$

- Háganse las transformaciones necesarias para eliminar la recursividad por la izquierda.
- Calcúlense los conjuntos de primeros y siguientes de cada no terminal.
- Compruébese que la gramática modificada cumple la condición LL(1).
- Constrúyase la tabla de análisis sintáctico LL(1) para esta nueva gramática.
- Finalmente, hágase la traza del análisis de la siguiente cadena, comprobando que las derivaciones son correctas mediante la construcción del árbol de análisis sintáctico.

```
begin
    var id,id: tipo
    var id:tipo
    fvar
end
    var id: tipo
theend
```

15. Dada la siguiente gramática:

$P \rightarrow D S$
 $D \rightarrow D V$
 $D \rightarrow \epsilon$
 $S \rightarrow S I$
 $S \rightarrow I$
 $V \rightarrow \text{decl id ;}$
 $V \rightarrow \text{decl id (} P \text{) ;}$
 $V \rightarrow \text{decl [} D \text{] id ;}$
 $I \rightarrow \text{id ;}$
 $I \rightarrow \text{begin } P \text{ end}$

- a. Háganse las transformaciones necesarias para que cumpla la condición LL(1).
- b. Constrúyase la tabla de análisis sintáctico LL(1) para esa nueva gramática.
- c. Hágase la traza de las cadenas:

```
decl id ( begin id ; )  
decl id ( decl [ decl id ; ] id ; ) ; id ;
```