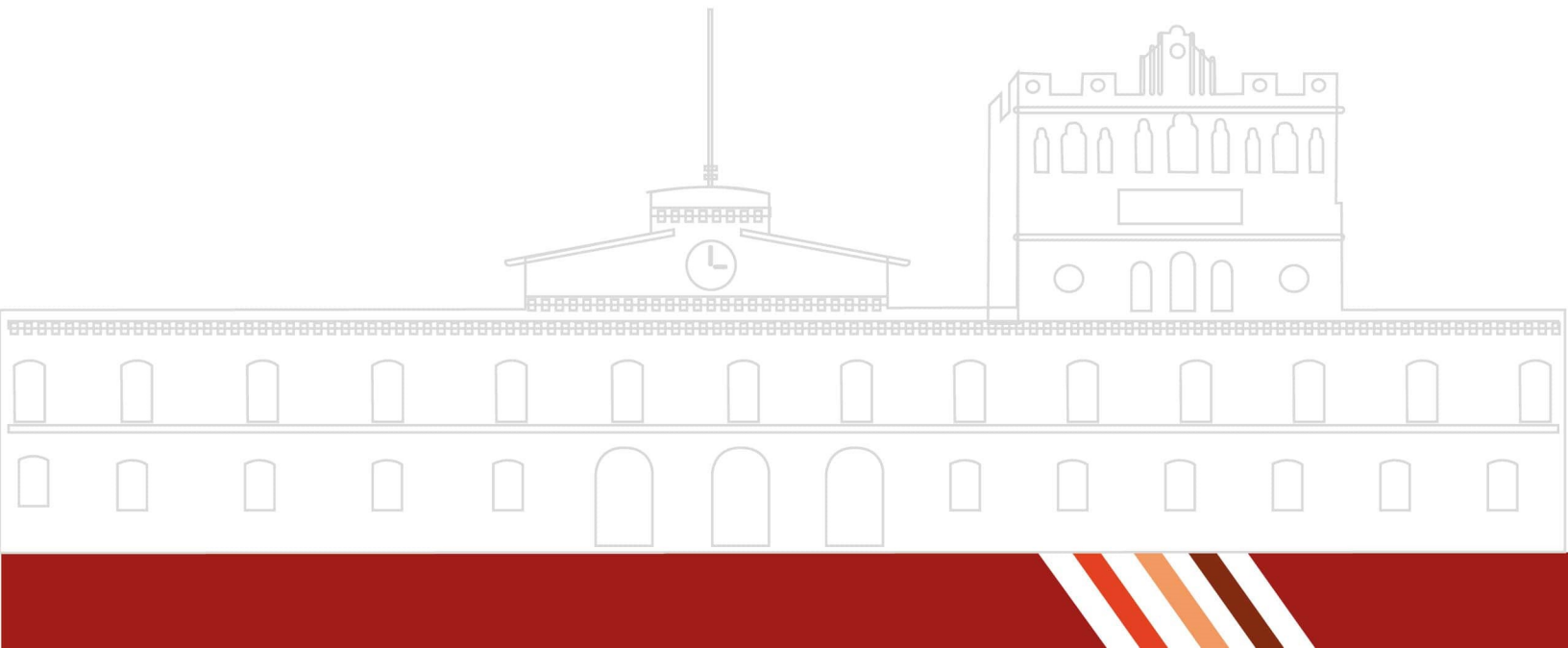


14 Ejercicios y actividades del capítulo 3

ALUMNO: Ian Alejandro Jaen Cruz
Dr. Eduardo Cornejo-Velázquez



Ejercicios

1. a) Escriba una gramática que genere el conjunto de cadenas $\{s; , s;s; , s;s;s; , \dots \}$.

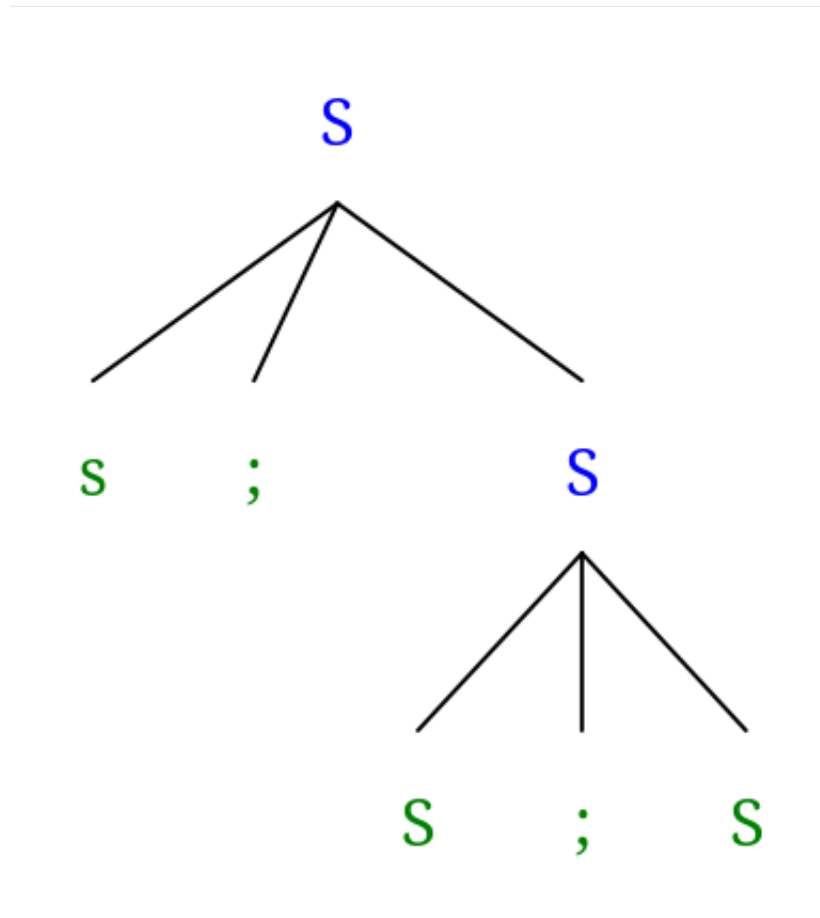


Figure 1: Arbol

2. Considere la siguiente gramática:

$\text{rexp} \rightarrow \text{rexp} \text{---} \text{rexp}$

— rexp rexp

— $\text{rexp} \text{---}^*$

— (rexp)

— letra

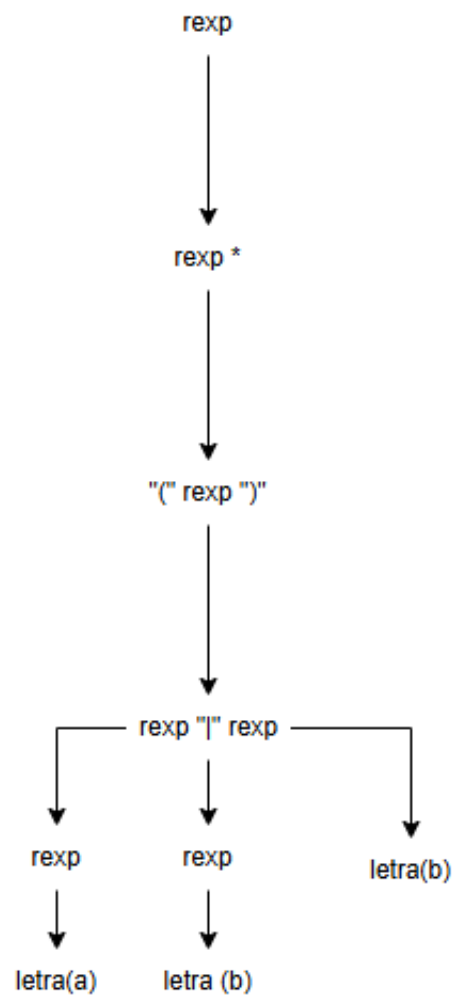


Figure 2: Arbol

3. De las siguientes gramáticas, describa el lenguaje generado por la gramática y genere árboles sintácticos con las respectivas cadenas.

a) $S \rightarrow S S + \mid S S * \mid a$ con la cadena $aa+a^*$.

b) $S \rightarrow 0 S 1 \mid 0 1$ con la cadena 000111 .

c) $S \rightarrow + S S \mid * S S \mid a$ con la cadena $+ * aaa$

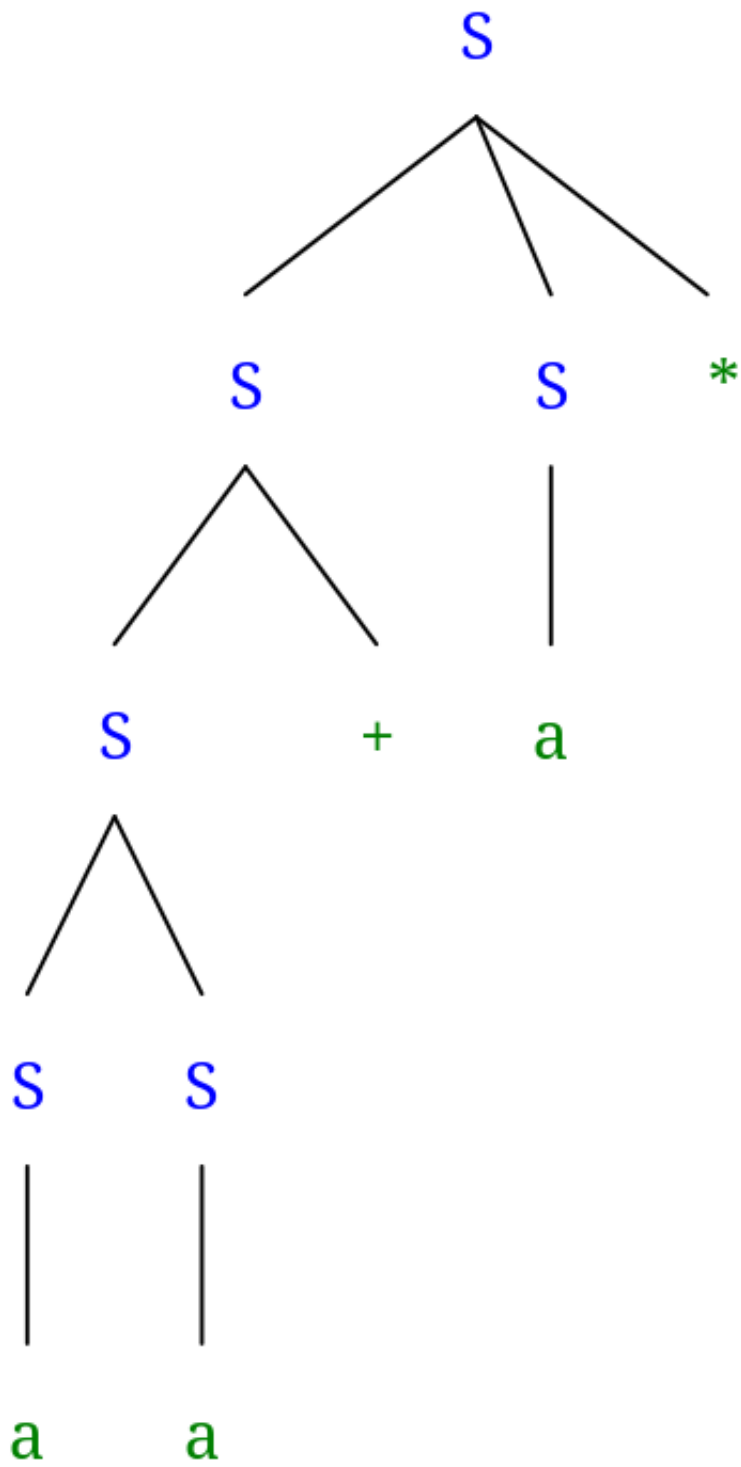


Figure 3: Arbol

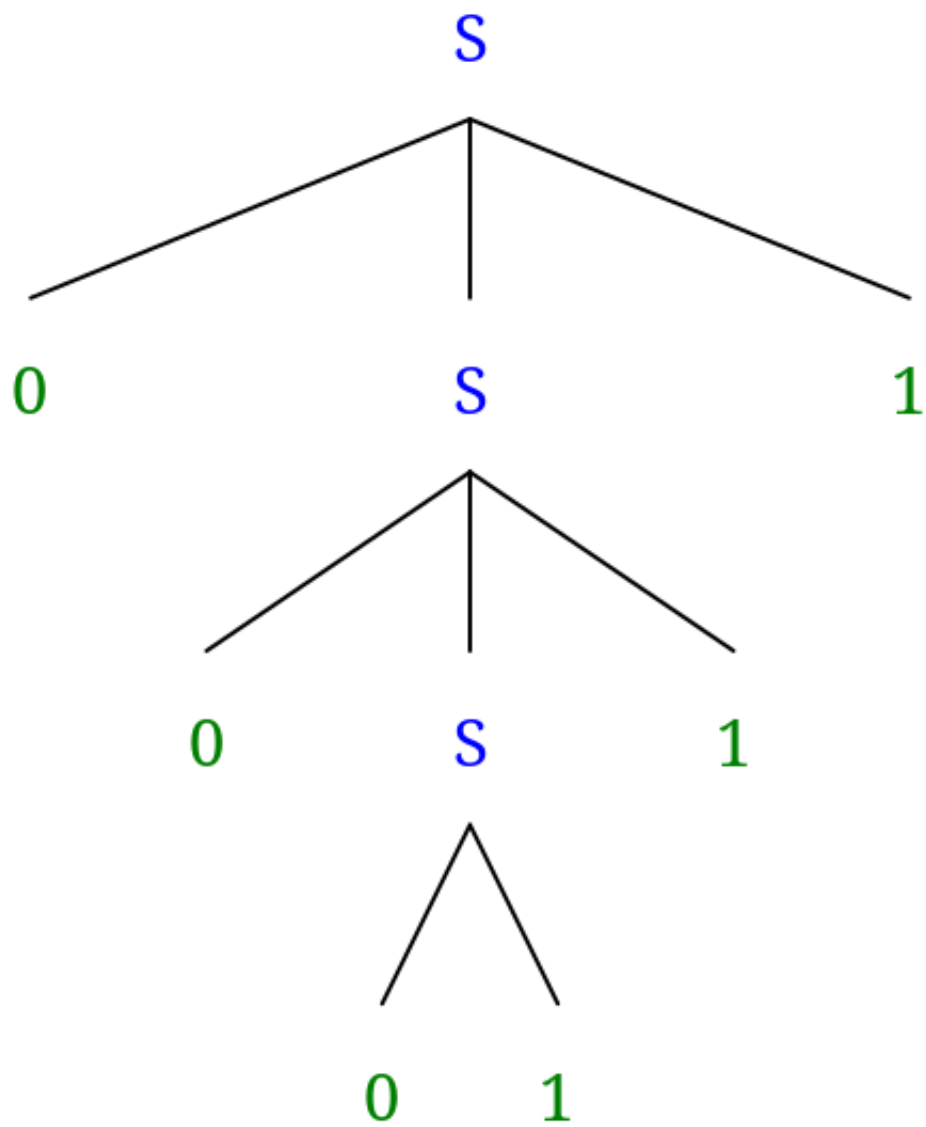


Figure 4: Arbol

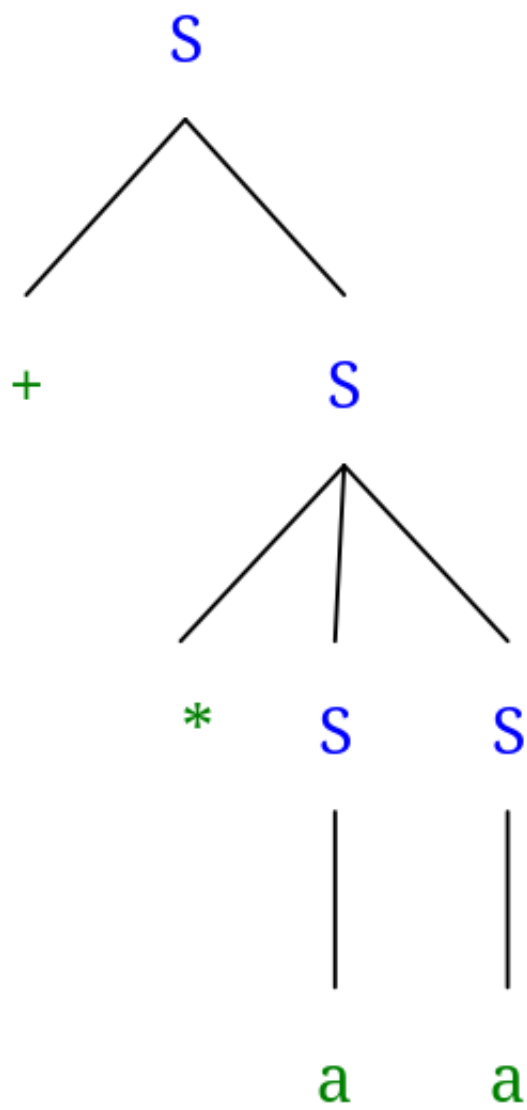


Figure 5: Arbol

4. ¿Cuál es el lenguaje generado por la siguiente gramática?

$S \rightarrow xSy \mid \epsilon$

Cadenas de la forma

$$x^n y^n$$

donde cada cadena contiene el mismo número de x y y , incluyendo la cadena vacía (ϵ).

5. Genere el árbol sintáctico para la cadena zazabzbz utilizando la siguiente gramática:

$S \rightarrow zMNz$

$M \rightarrow aNa$

$N \rightarrow bNb$

$N \rightarrow z$

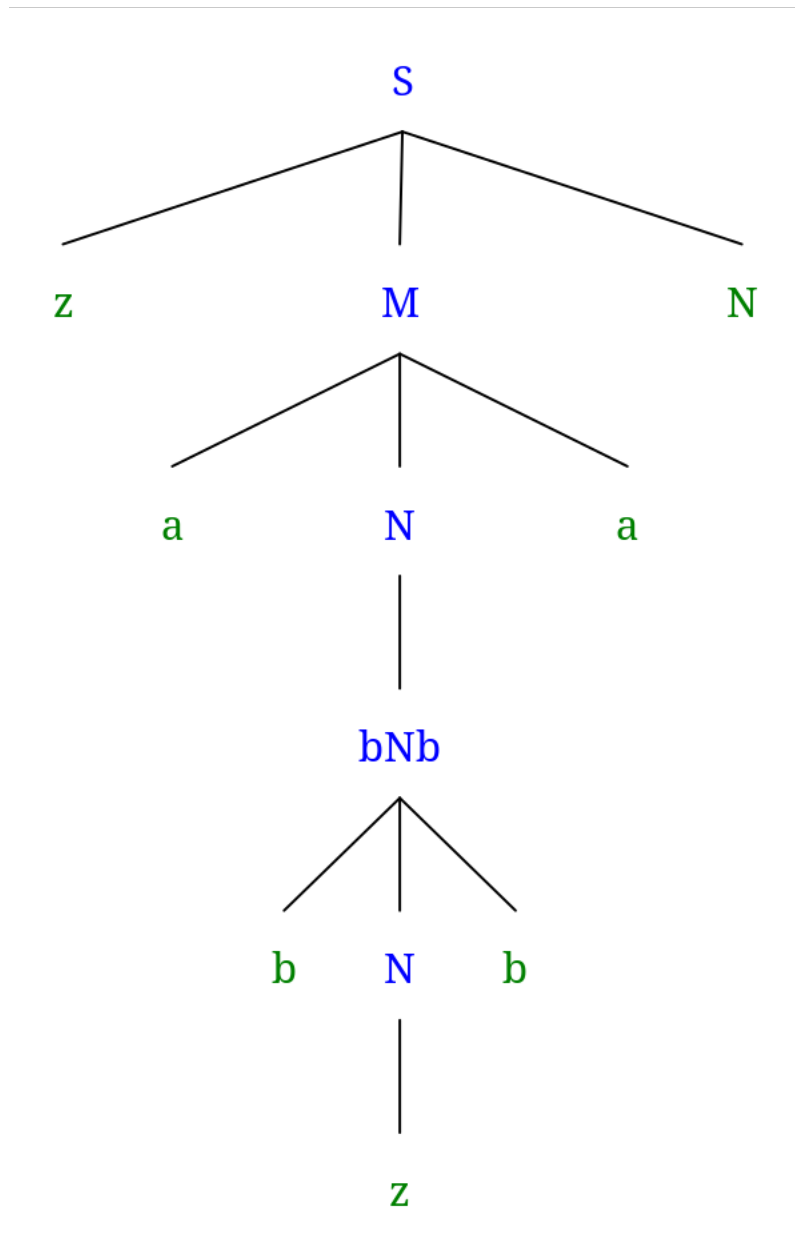


Figure 6: Arbol

6. Demuestre que la gramática que se presenta a continuación es ambigua, mostrando que la cadena ictictses tiene derivaciones que producen distintos árboles de análisis sintáctico.

$S \rightarrow \text{ict}S$
 $S \rightarrow \text{ictSe}S$
 $S \rightarrow s$

Derivación 1:

$S \rightarrow \text{ictSe}S \rightarrow \text{ictictSe}S \rightarrow \text{ictictses}S \rightarrow \text{ictSe}S \rightarrow \text{ictictSe}S \rightarrow \text{ictictses}$.

Derivación 2:

$S \rightarrow \text{ict}S \rightarrow \text{ictictSe}S \rightarrow \text{ictictses}S \rightarrow \text{ict}S \rightarrow \text{ictictSe}S \rightarrow \text{ictictses}$.

Los árboles difieren en la asociatividad del e.

7. Considere la siguiente gramática $S \rightarrow (L) \mid a$ $L \rightarrow L , S \mid S$ Encuéntrense árboles de análisis sintáctico para las siguientes frases:

- a) (a, a)
- b) (a, (a, a))
- c) (a, ((a, a), (a, a)))

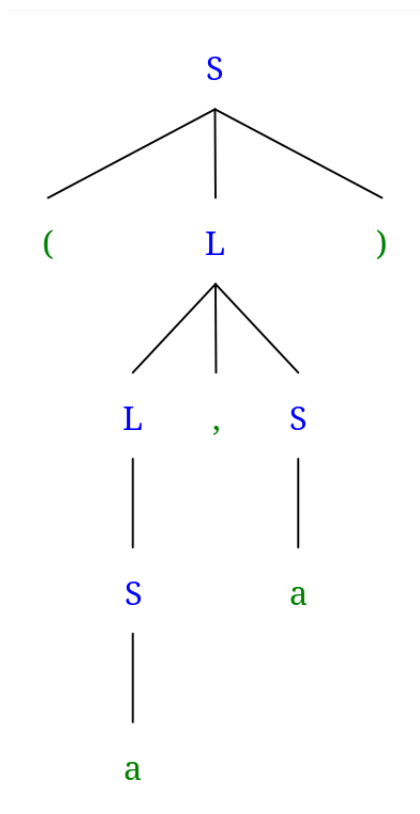


Figure 7: Arbol

8. Constrúyase un árbol sintáctico para la frase not (true or false) y la gramática:
bexpr \rightarrow bexpr or bterm — bterm
bterm \rightarrow bterm and bfactor — bfactor
bfactor \rightarrow not bfactor — (bexpr) — true — false

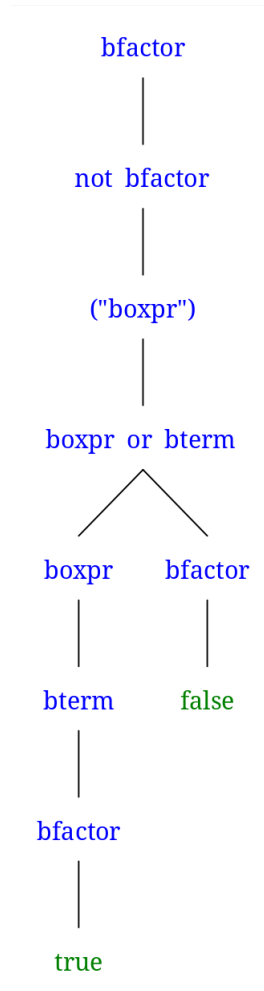


Figure 8: Arbol

9. Diseñe una gramática para el lenguaje del conjunto de todas las cadenas de símbolos 0 y 1 tales que todo 0 va inmediatamente seguido de al menos un 1.

$$S \rightarrow 1S01S$$

10. Elimine la recursividad por la izquierda de la siguiente gramática:

$$S \rightarrow (L) \mid a$$

$$L \rightarrow L, S \mid S$$

$$S \rightarrow (L) \mid a$$

$$L \rightarrow SL'$$

$$L' \rightarrow, SL' \mid \varepsilon$$

13. La gramática 3.2, sólo maneja las operaciones de suma y multiplicación, modifique esa gramática para que acepte, también, la resta y la división; Posteriormente, elimine la recursividad por la izquierda de la gramática completa y agregue la opción de que F, también pueda derivar en num, es decir, $F \rightarrow (E) \mid id \mid num$

Para eliminar la recursión por la izquierda, la gramática se transforma en:

$$E \rightarrow TE'$$

$$E' \rightarrow +TE' \mid -TE' \mid \varepsilon$$

$$T \rightarrow FT'$$

$$T' \rightarrow *FT' \mid /FT' \mid \varepsilon$$

$$F \rightarrow (E) \mid id \mid num$$