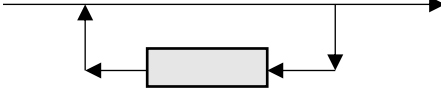

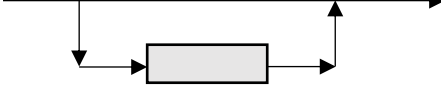


## Práctica 2

**Ejercicio 1: Complete el siguiente cuadro:**

Meta-símbolos utilizados por		Símbolo utilizado en Diagramas sintácticos	Significado
BNF	EBNF		
Palabra terminal	Palabra terminal	Ovalo	Definición de un elemento terminal.
Digito	Digito	Rectángulo	Definición de un elemento no terminal
::=	::=	Diagrama con rectángulos, óvalos y flechas	Meta-símbolo de definición que indica que el elemento a su izquierda se puede definir según el esquema de la derecha
	( )	Flecha que se divide en dos o más caminos	Meta-símbolo de opción que indica que puede elegirse uno y solo uno de los meta símbolos
< p > < p1 >	{ }		Repetición
	*		Repetición de 0 o más veces.
	+		Repetición de 1 o más veces.
	[ ]		Elemento optativo (puede o no estar).

*Nota: p y p1 son producciones simbólicas.*

**Ejercicio 2: ¿Cuál es la importancia de la sintaxis para un lenguaje? ¿Cuáles son sus elementos?**

La sintaxis de un lenguaje de programación es la estructura en que se organizan los distintos elementos sintácticos, como espacios, identificadores, operadores, etc. Es decir, el orden que tienen unos con respecto a otros.

Una sintaxis se evalúa según varios criterios: que sea fácil de leer, de escribir, de verificar (chequear en busca de errores), fácil de traducir y que carezca de ambigüedad. Esta última significa que un

mismo código puede tener 2 o más interpretaciones o traducciones posibles, en cuyo caso no se puede decidir que alternativa elegir.

Es muy importantes ya que establece las reglas para que el programador se comunice con el procesador de forma correcta sintácticamente.

#### Elementos de la sintaxis

- Alfabeto o conjunto de caracteres
- Identificadores
- Operadores
- Palabra clave y palabra reservada
- Comentarios y uso de blancos

#### **Ejercicio 3: ¿Explique a qué se denomina regla lexicográfica y regla sintáctica?**

- Reglas léxicas: Conjunto de reglas para formar las “word”, a partir de los caracteres del alfabeto
- Reglas sintácticas: Conjunto de reglas que definen como formar las “expresiones” y “sentencias”

#### **Ejercicio 4: ¿En la definición de un lenguaje, a qué se llama palabra reservadas? ¿A qué son equivalentes en la definición de una gramática? De un ejemplo de palabra reservada en el lenguaje que más conoce. (Ada,C,Ruby,Python,..)**

Palabra reservada, son palabras claves (palabras que tienen significado dentro de un contexto) que además no pueden ser usadas por el programador como identificador de otra entidad. Equivale al conjunto de símbolos terminales de la gramática.

#### **Ejercicio 5: Dada la siguiente gramática escrita en BNF:**

$G = (N, T, S, P)$

$N = \{ \langle \text{numero\_entero} \rangle, \langle \text{digito} \rangle \}$

$T = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

$S = \langle \text{numero\_entero} \rangle$

$P = \{$

$\langle \text{numero\_entero} \rangle ::= \langle \text{digito} \rangle \langle \text{numero\_entero} \rangle \mid \langle \text{numero\_entero} \rangle \langle \text{digito} \rangle \mid \langle \text{digito} \rangle$

$\langle \text{digito} \rangle ::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

$\}$

**a- Identifique las componentes de la misma**

$G$  = 4-tupla de Conjunto de reglas finita que define un conjunto infinito de posibles sentencias válidas en el lenguaje.

N = Conjunto de símbolos no terminales

T = Conjunto de símbolos terminales

S = Símbolo distinguido de la gramática que pertenece a N

P = Conjunto de producciones

**b- Indique porqué es ambigua y corrijala**

Es ambigua por  $\langle \text{digito} \rangle \langle \text{numero\_entero} \rangle \mid \langle \text{numero\_entero} \rangle \langle \text{digito} \rangle$

La asociativa es por la izquierda o por la derecha, en este caso se estarían generando DOS arboles de derivación

CORRECCION

$G = (N, T, S, P)$

$N = \{ \langle \text{numero\_entero} \rangle, \langle \text{digito} \rangle \}$

$T = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

$S = \langle \text{numero\_entero} \rangle$

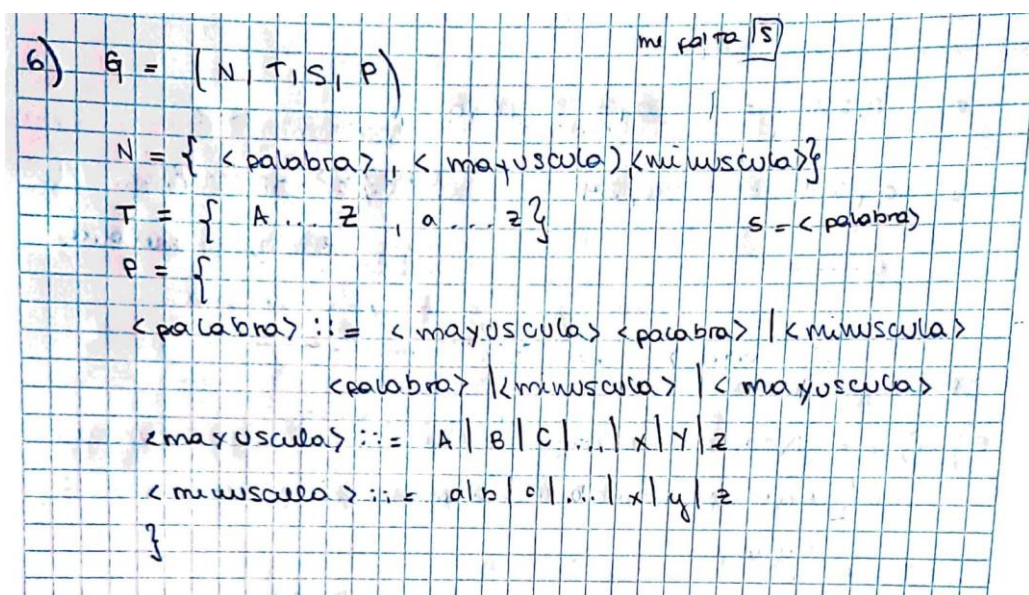
$P = \{$

$\langle \text{numero\_entero} \rangle ::= \langle \text{digito} \rangle \langle \text{numero\_entero} \rangle \mid \langle \text{digito} \rangle$

$\langle \text{digito} \rangle ::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

$\}$

**Ejercicio 6: Defina en BNF (Gramática de contexto libre desarrollada por Backus- Naur) la gramática para la definición de una palabra cualquiera.**



Ejercicio 7: Defina en EBNF la gramática para la definición de números reales. Inténtelo desarrollar para BNF y explique las diferencias con la utilización de la gramática EBNF.

4)  $G = (N, T, S, P)$  EBNF  $S = \langle \text{numero\_real} \rangle$

$N = \{ \langle \text{numero\_real} \rangle, \langle \text{digito} \rangle \}$

$T = \{ '0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '+', '-', '.', ',' \}$

$P = \{$

$\langle \text{numero\_real} \rangle ::= [(+|-)] \langle \text{digito} \rangle^+ [, \langle \text{digito} \rangle^+]$

$\langle \text{digito} \rangle ::= 0|1|2|3|4|5|6|7|8|9$

$\}$

$N = \{ \langle \text{numero\_real} \rangle, \langle \text{decimal} \rangle, \langle \text{digito} \rangle, \langle \text{numero\_entero} \rangle, \langle \text{numero\_entero\_neg} \rangle \}$

$T = \{ '0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '+', '-', '.', ',' \}$

$P = \{$

$\langle \text{numero\_real} \rangle ::= \langle \text{numero\_entero\_neg} \rangle | \langle \text{numero\_entero} \rangle$

$\langle \text{decimal} \rangle$

$\langle \text{numero\_entero\_neg} \rangle ::= + \langle \text{numero\_entero} \rangle |$

$- \langle \text{numero\_entero} \rangle | \langle \text{numero\_entero} \rangle$

$\langle \text{numero\_entero} \rangle ::= \langle \text{digito} \rangle | \langle \text{digito} \rangle \langle \text{numero\_entero} \rangle$

$\langle \text{decimal} \rangle ::= , \langle \text{numero\_entero} \rangle$

$\langle \text{digito} \rangle ::= 0|1|2|3|4|5|6|7|8|9$

$\}$

$S = \langle \text{numero\_real} \rangle$

↓

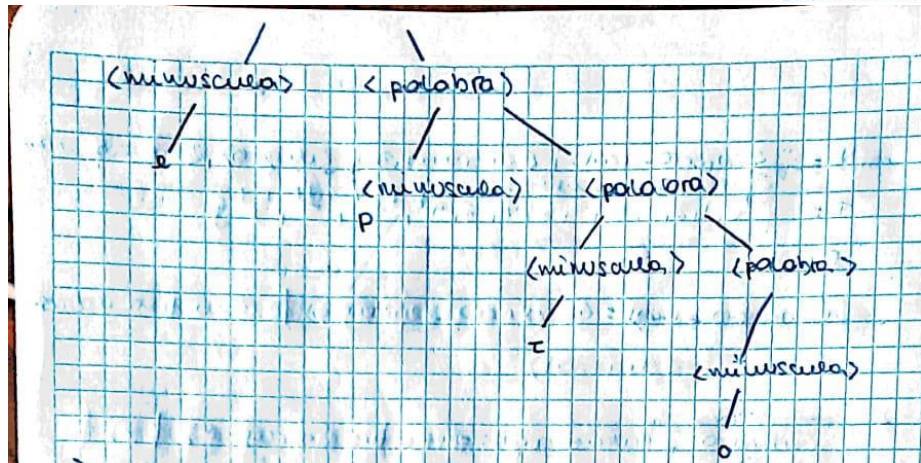
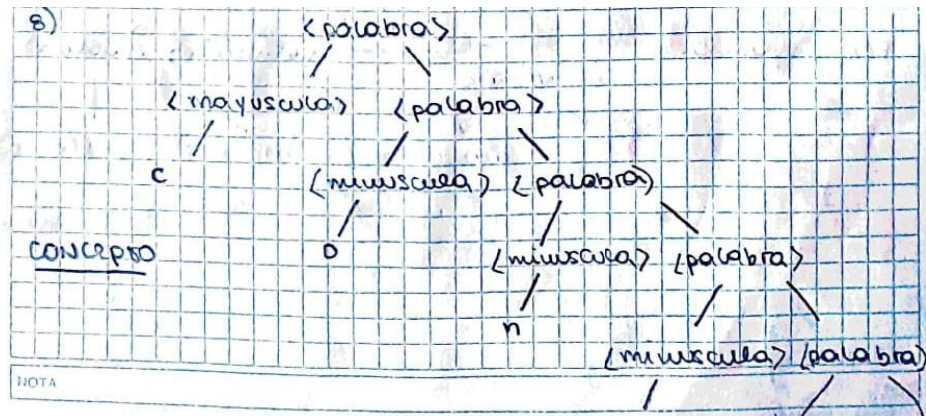
preguntas si o no

Se puede observar como BNF no es tan claro de leer y a su vez hay que implementar muchos cambios para en EBNF se pueden hacer con más facilidad. A su vez se requiere definir más no terminales que en EBNF no hay para

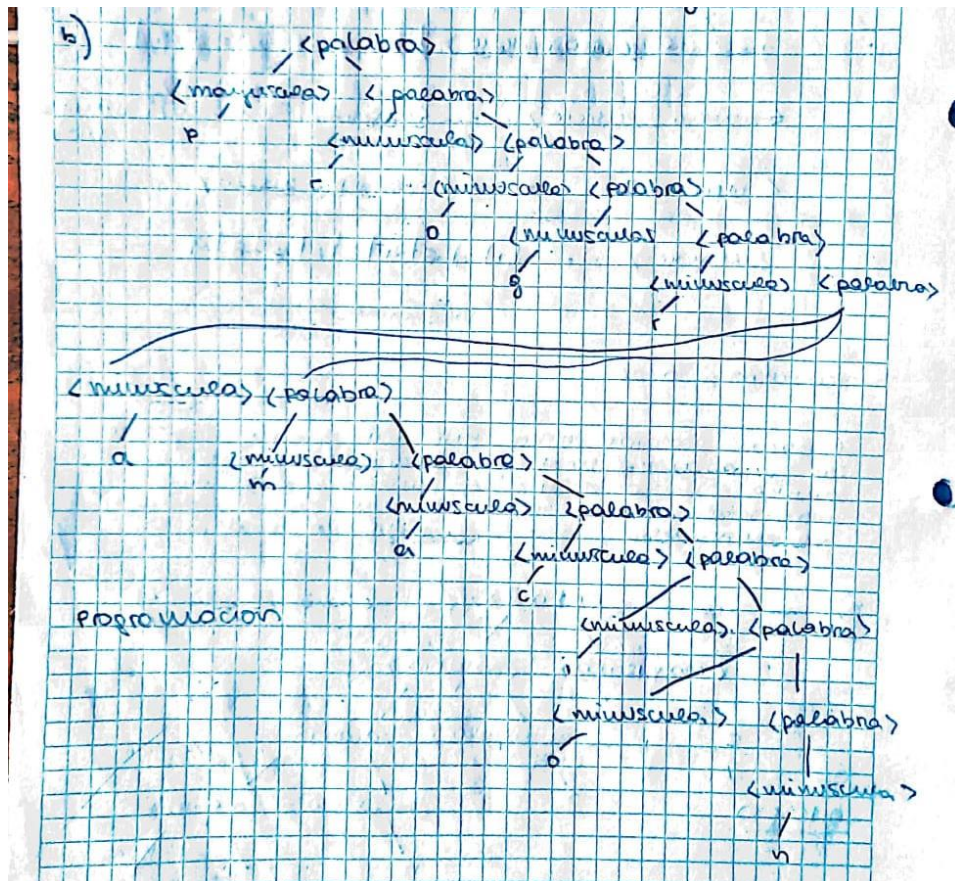
Ejercicio 8: Utilizando la gramática que desarrolló en los puntos 6 y 7, escriba el árbol sintáctico de:

a. Conceptos



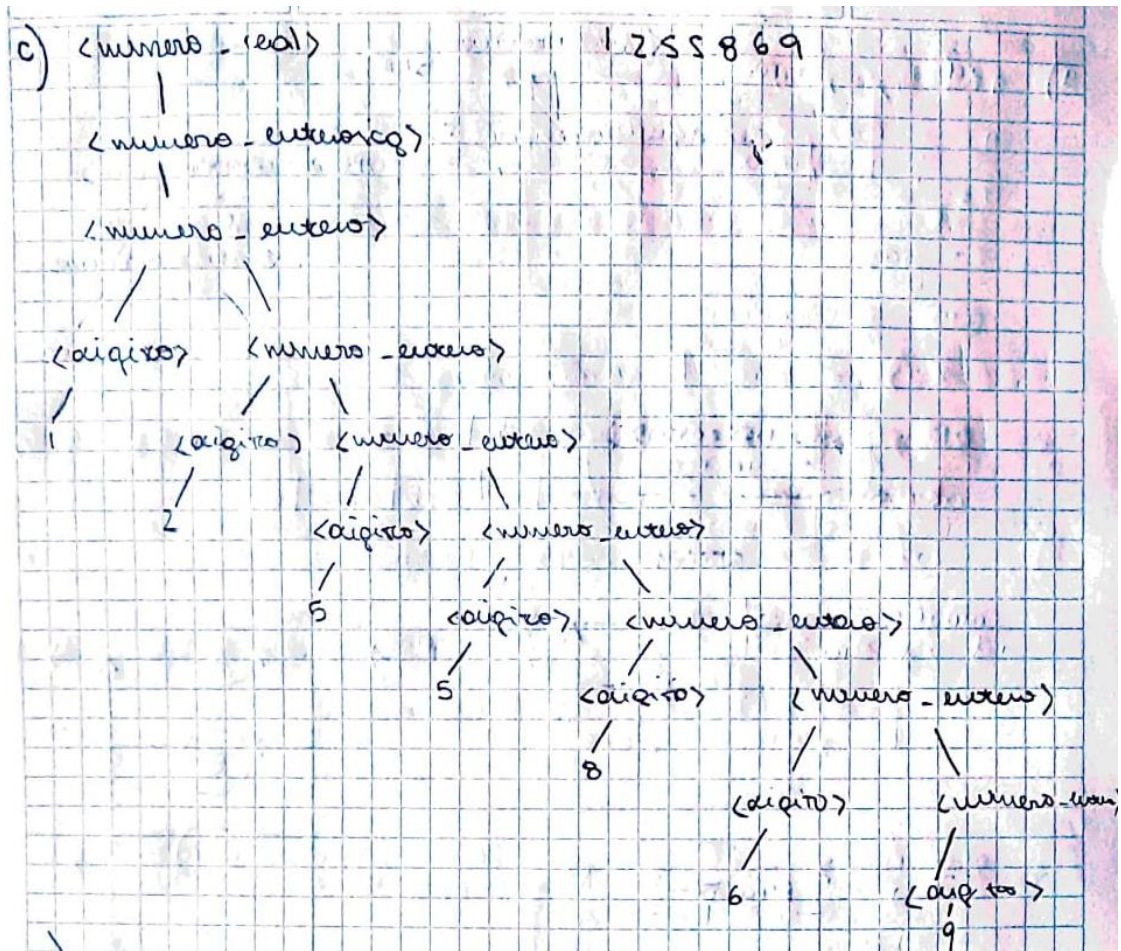


## b. Programación

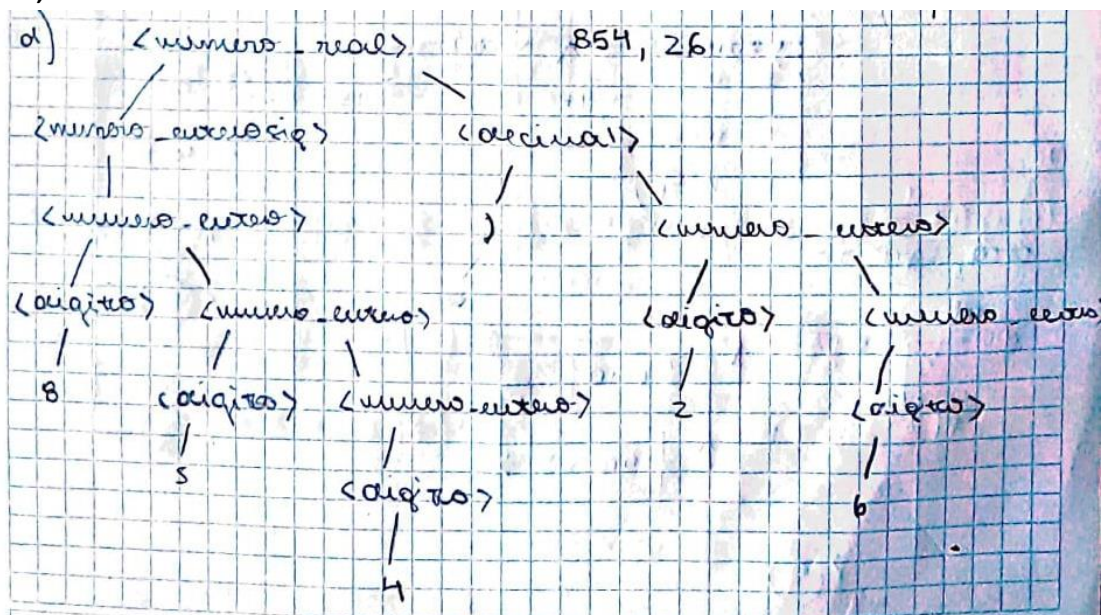




c. 1255869



d. 854,26

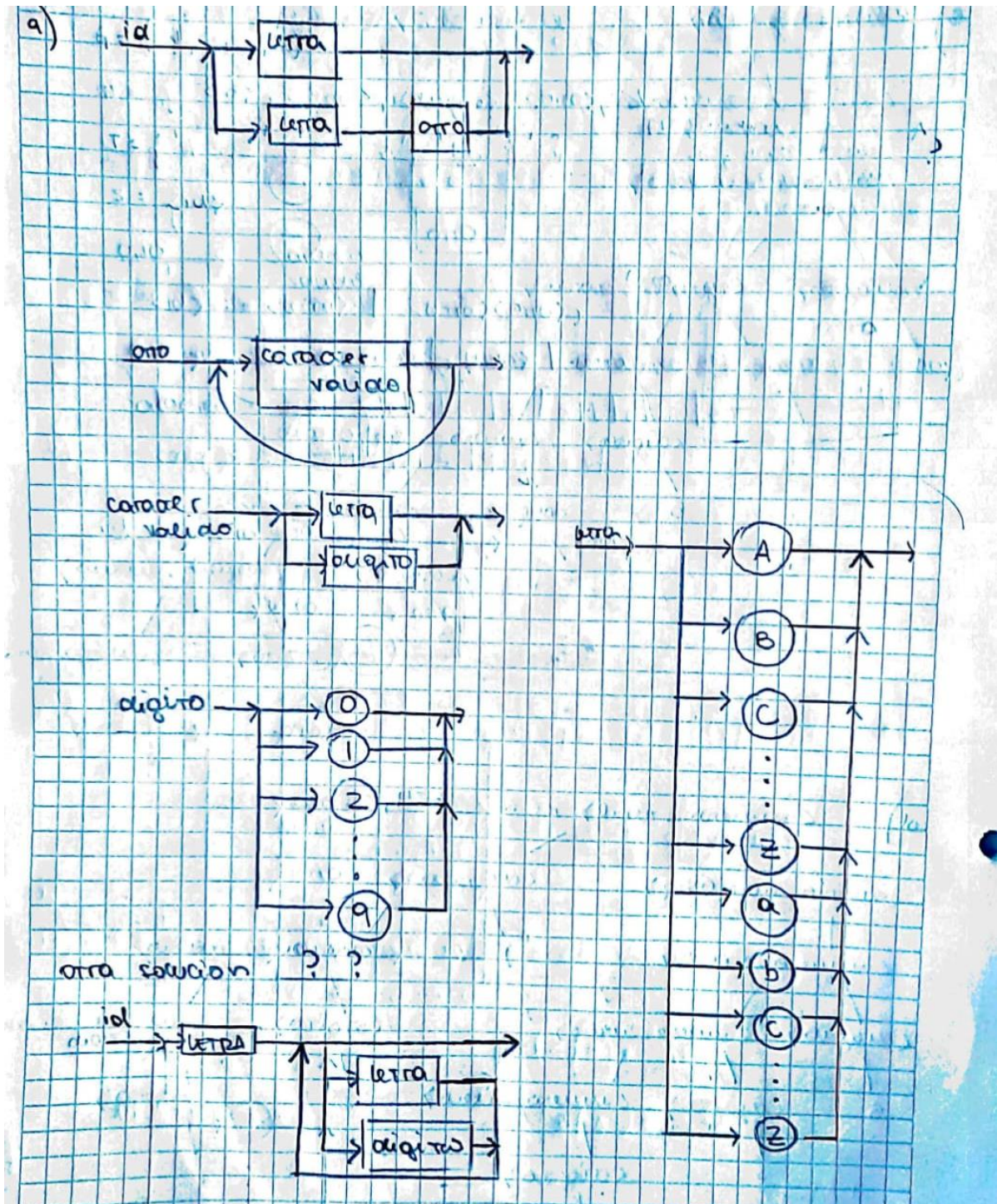


e. Conceptos de lenguajes

Con la gramática definida en el punto 6 y 7 no se puede representar un String.



Ejercicio 9: Defina utilizando diagramas sintácticos la gramática para la definición de un identificador de un lenguaje de programación. Tenga presente como regla que un identificador no puede comenzar con números.





# Ejercicio 10:

- a) Defina con EBNF la gramática para una expresión numérica, donde intervienen variables y números. Considerar los operadores +, -, \* y / sin orden de prioridad. No considerar el uso de paréntesis.

a)

$$G = (N, T, S, P)$$

$$N = \{ \langle \text{exp} \rangle, \langle \text{digito} \rangle, \langle \text{id} \rangle, \langle \text{operador} \rangle, \langle \text{letra} \rangle, \langle \text{nro} \rangle \}$$

$$T = \{ 'A' \dots 'Z', 'a' \dots 'z', '0' \dots '9', '+', '-', '*', '/' \}$$

$$S = \langle \text{exp} \rangle$$

$$P = \{$$

$$\langle \text{exp} \rangle ::= (\langle \text{id} \rangle | \langle \text{nro} \rangle) \{ \langle \text{operador} \rangle (\langle \text{id} \rangle | \langle \text{nro} \rangle) \}^+$$

$$\langle \text{digito} \rangle ::= 0 | 1 | 2 | \dots | 9$$

$$\langle \text{operador} \rangle ::= (+ | - | * | /)$$

$$\langle \text{id} \rangle ::= \langle \text{letra} \rangle \{ (\langle \text{letra} \rangle | \langle \text{digito} \rangle) \}^*$$

$$\langle \text{nro} \rangle ::= A | B | C | \dots | Z | a | b | c | \dots | z$$

$$\langle \text{nro} \rangle ::= [1-] \{ \langle \text{digito} \rangle \}^+$$

- b) A la gramática definida en el ejercicio anterior agregarle prioridad de operadores.

b)

$$G = (N, T, S, P)$$

$$N = \{ \langle \text{exp} \rangle, \langle \text{exp con prioridad} \rangle, \langle \text{operador sin prioridad} \rangle, \langle \text{operador con prioridad} \rangle, \langle \text{digito} \rangle, \langle \text{id} \rangle, \langle \text{letra} \rangle, \langle \text{nro} \rangle \}$$

$$T = \{ 'A' \dots 'Z', 'a' \dots 'z', '0' \dots '9', '+', '-', '*', '/' \}$$

$$S = \langle \text{exp} \rangle$$

$$P = \{$$

$$\langle \text{exp} \rangle ::= \langle \text{exp con prioridad} \rangle \{ \langle \text{operador sin prioridad} \rangle \langle \text{exp con prioridad} \rangle \}^*$$

$$\langle \text{exp con prioridad} \rangle ::= [ + | - ] (\langle \text{id} \rangle | \langle \text{nro} \rangle) \{ \langle \text{operador con prioridad} \rangle (\langle \text{id} \rangle | \langle \text{nro} \rangle) \}^*$$

$$\langle \text{operador con prioridad} \rangle ::= ( * | / )$$

$$\langle \text{operador sin prioridad} \rangle ::= ( + | - )$$

$$\langle \text{digito} \rangle ::= 0 | 1 | 2 | \dots | 9$$

$$\langle \text{id} \rangle ::= \langle \text{letra} \rangle \{ (\langle \text{letra} \rangle | \langle \text{digito} \rangle) \}^*$$

$$\langle \text{nro} \rangle ::= A | B | C | \dots | Z | a | b | c | \dots | z$$

$$\langle \text{nro} \rangle ::= [1-] \{ \langle \text{digito} \rangle \}^+$$



- c) Describa con sus palabras los pasos y decisiones que tomó para agregarle prioridad de operadores al ejercicio anterior.

Una operación si prioridad está compuesta por operaciones con prioridad (separó en términos)

**Ejercicio 11:** La siguiente gramática intenta describir sintácticamente la sentencia for de ADA, indique cuál/cuáles son los errores justificando la respuesta.

N= {<sentencia\_for>, <bloque>, <variable>, <letra>, <cadena>, <digito>, <otro>, <operacion>, <llamada\_a\_funcion>, <numero>, <sentencia> }

P= {

<sentencia\_for>::= for (i= IN 1..10) loop <bloque> end loop;

<variable>::= <letra> | <cadena>

<cadena>::= { ( <letra> | <digito> | <otro> ) }+

<letra>::= ( a | .. | z | A | .. | Z )

<digito>::= ( 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 )

<bloque>::= <sentencia> | <sentencia> <bloque> | <bloque> <sentencia> ;

<sentencia>::= <sentencia\_asignacion> | <llamada\_a\_funcion> | <sentencia\_if> |

<sentencia\_for> | <sentencia\_while> | <sentencia\_switch> }

La gramática es una 4-tupla, en este caso no están definidas la T y la S, es decir, faltan poner los terminales que se utilizan y el símbolo distinguido (la gramática que se está definiendo).

A su vez, faltan agregar en N <sentencia\_if>, <sentencia\_while>, <sentencia\_switch>.

En caso de que las sentencias que se utilizan no estén definidas aparte (sub-gramatica), habría que realizar la producción de ellas acá (tales como <otro>, <sentencia\_asignacion>, <llamada\_a\_funcion>, <sentencia\_if>, <sentencia\_for>, <sentencia\_while>, <sentencia\_switch>, <operacion>, <numero>)

<bloque>::= <sentencia> | <sentencia> <bloque> | <bloque> <sentencia> ; Es ambiguo

**Ejercicio 12:** Realice en EBNF la gramática para la definición un tag div en html 5. (Puede ayudarse con el siguiente enlace (<https://developer.mozilla.org/es/docs/Web/HTML/Elemento/div>))

$\langle \text{sentencia} \rangle ::= \langle \text{if} \rangle ; | \langle \text{while} \rangle | \langle \text{for} \rangle ; | \dots ;$

HOJA N°

FECHA

CP1P

EBNF

12)  $G = \{ N, T, S, P \}$

$P = \{ \langle \text{tag-div} \rangle ::= \langle \text{div apertura} \rangle [ \langle \text{bloque} \rangle ] \langle \text{div cierre} \rangle$

$\langle \text{div apertura} \rangle ::= \langle \text{div} \text{farrueto} \rangle^* \rangle$

$\langle \text{div cierre} \rangle ::= \langle / \text{div} \rangle$

$\langle \text{atributo} \rangle ::= \langle \text{palabra} \rangle = \langle \text{cadena} \rangle$

$\langle \text{palabra} \rangle ::= \{ \langle \text{letra} \rangle \}^+$

$\langle \text{cadena} \rangle ::= \{ \langle \text{letra} \rangle | \langle \text{numero} \rangle | \langle \text{caracteres especiales} \rangle \}^+$

$\langle \text{letra} \rangle ::= \{ A | B | C | \dots | Z | a | b | c | \dots | z \}$

$\langle \text{numero} \rangle ::= \{ 0 | 1 | 2 | \dots | 9 \}$

$\langle \text{caracter especial} \rangle ::= \{ + | - | / | * | \dots | | ; | : \}$

$\langle \text{bloque} \rangle ::= [ \langle \text{cadena} \rangle | \langle \text{tag-div} \rangle | \langle \text{tag-span} \rangle | \dots ]$

tenemos  
en  
cadena  
si  
esto  
va  
bien  
}

$\langle \text{tag-span} \rangle ::= \langle \text{span} \{ \text{atributo} \}^* \rangle [ \langle \text{bloque} \rangle ] \langle / \text{span} \rangle$

esta  
OK  
asumiendo  
que

$N = \{ \langle \text{tag-div} \rangle, \langle \text{div apertura} \rangle, \langle \text{bloque} \rangle, \langle \text{div cierre} \rangle,$

$\langle \text{carrietos} \rangle, \langle \text{palabra} \rangle, \langle \text{cadena} \rangle, \langle \text{letra} \rangle,$

$\langle \text{numero} \rangle, \langle \text{caracter especial} \rangle, \langle \text{tag-span} \rangle \}$

$T = \{ 'A' \dots 'Z', 'a' \dots 'z', '0' \dots '9', '+', \dots, '<', '>', '/' \}$

$S = \{ \langle \text{tag-div} \rangle \}$

los espacios y  
cadenas  
de

línea  
no se  
tienen  
en  
cadena

tema que tiene  
en cadena  
si es y;  
o una  
regla  
especial

los puntos  
en los  
no se  
pero  
no los  
algunos

especial  
en las reglas

NOTA



**Ejercicio 13:** Defina en EBNF una gramática para la construcción de números primos. ¿Qué debería agregar a la gramática para completar el ejercicio?

La construcción es igual que la de los números enteros. Para completar el ejercicio se debería agregar la semántica para verificar si es un número primo o no.

**Ejercicio 14:** Sobre un lenguaje de su preferencia escriba en EBNF la gramática para la definición de funciones o métodos o procedimientos (considere los parámetros en caso de ser necesario)

14)  $G = (N, T, S, P)$  JAVASCRIPT función → igual a otros y flecha

$N = \{ \langle \text{función normal} \rangle, \langle \text{letra minúscula} \rangle, \langle \text{letra mayúscula} \rangle, \langle \text{código} \rangle, \langle \text{parámetro} \rangle, \langle \text{nombre función} \rangle, \langle \text{cuerpo} \rangle, \langle \text{return} \rangle, \langle \text{cuerpo} \rangle \}$

$T = \{ '0'.. '9', 'a'.. 'z', 'A'.. 'Z', '-', '.', '}' \}$

$S = \langle \text{función normal} \rangle$

$P = \{$

- $\langle \text{id} \rangle ::= \langle \text{cadena min} \rangle \{ \langle \text{cadena min} \rangle | \langle \text{código} \rangle \}^*$
- $\langle \text{nombre función} \rangle ::= \langle \text{id} \rangle \langle \text{parámetro} \rangle$
- $\langle \text{cuerpo} \rangle ::= \{ \langle \text{sentencia} \rangle^* [ \langle \text{return} \rangle ] \}$
- $\langle \text{función normal} \rangle ::= \text{función } \langle \text{nombre función} \rangle \langle \text{cuerpo} \rangle$
- $\langle \text{letra minúscula} \rangle ::= (a|b|c|...|z)$
- $\langle \text{letra mayúscula} \rangle ::= (A|B|C|...|Z)$
- $\langle \text{código} \rangle ::= (0|1|2|...|9)$
- $\langle \text{parámetro} \rangle ::= \{ \langle \text{id} \rangle, \}^*$
- $\langle \text{cadena min} \rangle ::= (\langle \text{letra mayúscula} \rangle | \langle \text{letra minúscula} \rangle | -)$

supongo que  $\langle \text{sentencia} \rangle$  y  $\langle \text{return} \rangle$  son subgramáticas ya definidas.

preguntar

NOTACIONES:

- No hace falta aclarar los espacios o saltos de línea
- Cuando se trata de una sentencia, puedo poner (sentencia\_if| sentencia\_while|...) sin la necesidad de definir las