



## Gramática.

$G = \{\{\text{Terminales}\}, \{\text{No terminales } \langle \text{Nombre} \rangle\}, S, \text{Inicio}, \{\text{Reglas de producción}\}\}$

$G = \{\{a, b, c, d, e, f, \dots, z, \langle, \rangle, \langle =, \rangle =, =, ==, !, !=, |, ||, ^, \%, \&\&, +, -, *, /, (), ., ;, :, 0 \dots 9, \text{“ ”, coma, } \{ \}, [ ], : \text{ inicioclase, finclase, iniciodeclararvar, findeclararvar, finper, finsi, fincontrario, finmien}, \{ \dots \}, \langle \text{Programa} \rangle\}\}$

$\langle \text{op\_aritmético} \rangle :: + | - | * | / | \% | ^$

$\langle \text{operando} \rangle :: \langle \text{constante} \rangle | \backslash \langle \text{ID} \rangle \backslash$

$\langle \text{operación} \rangle :: \langle \text{operando} \rangle \langle \text{op\_aritmético} \rangle \langle \text{operando} \rangle$

$\langle \text{asignación} \rangle :: \backslash \langle \text{ID} \rangle \backslash = \langle \text{operando} \rangle | \backslash \langle \text{ID} \rangle \backslash = \langle \text{operación} \rangle ;$

$\langle \text{asignar} \rangle :: \langle \text{asignación} \rangle | \langle \text{asignar} \rangle \langle \text{asignación} \rangle$

$\langle \text{etiqueta} \rangle :: \text{iniciodeclararvar} | \text{findeclararvar} | \text{finper} | \text{finsi} | \text{fincontrario} | \text{finmien}$

$\langle \text{contrario} \rangle :: \text{contrario}$

$\langle \text{mien} \rangle :: \text{mien} (\langle \text{cond} \rangle)$

$\langle \text{op\_log} \rangle :: || | \&\&$

$\langle \text{cond} \rangle :: \langle \text{condición} \rangle | \langle \text{cond} \rangle \langle \text{oplog} \rangle \langle \text{condición} \rangle // \text{esta es condición compleja}$

$\langle \text{si} \rangle :: \text{si} (\langle \text{cond} \rangle)$

$\langle \text{op\_condicional} \rangle :: < | > | < = | > = | == | !=$

$\langle \text{op\_condicionalper} \rangle :: ::$

$\langle \text{condición} \rangle :: \langle \text{operando} \rangle \langle \text{opcondicional} \rangle \langle \text{operando} \rangle$

$\langle \text{opdec} \rangle :: ++ | --$

$\langle \text{condición\_per} \rangle :: \backslash \langle \text{ID} \rangle \backslash \langle \text{op\_condicionalper} \rangle (\backslash \langle \text{ID} \rangle \backslash | \langle \text{numént} \rangle | \langle \text{numflot} \rangle)$

$\langle \text{per} \rangle :: \text{per} (\backslash \langle \text{ID} \rangle \backslash = (\backslash \langle \text{ID} \rangle \backslash | \langle \text{numént} \rangle | \langle \text{numflot} \rangle) ; \langle \text{condición\_per} \rangle ; \backslash \langle \text{ID} \rangle \backslash \langle \text{opdec} \rangle)$

$\langle \text{leer} \rangle :: \text{leer} | \backslash \langle \text{ID} \rangle \backslash ;$

$\langle \text{constante} \rangle :: \text{“} \langle \text{cadena} \rangle \text{”} | \langle \text{numént} \rangle | \langle \text{numflot} \rangle$

$\langle \text{imprimir} \rangle :: \text{sout} (\langle \text{operando} \rangle) ;$

$\langle \text{sentencia} \rangle :: \langle \text{imprimir} \rangle | \langle \text{leer} \rangle | \langle \text{per} \rangle | \langle \text{si} \rangle | \langle \text{mien} \rangle | \langle \text{contrario} \rangle | \langle \text{etiqueta} \rangle | \langle \text{asignar} \rangle$



<sentencias> :: <sentencia> | <sentencias> <sentencia>

<num> :: 1 | 2 | 3 | 4 | ... | 9

<nument> :: <num> | <nument> <num>

<numflot> :: <nument>.< nument>

<espacio> :: /s

<vacio> ::

<mayuscula> :: A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U |  
V | W | X | Y | Z

<sim> :: <minuscula> | <mayuscula> | <num> | <vacio> | <espacio>

<cadena> :: <sim> | <cadena> <sim>

<dato> :: “ <cadena> ” | <nument> | <numflot>

<tipodedato> :: ent | cad | flot

<minuscula> :: a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x |  
y | z

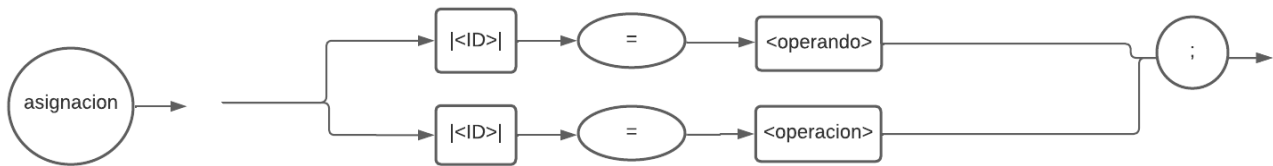
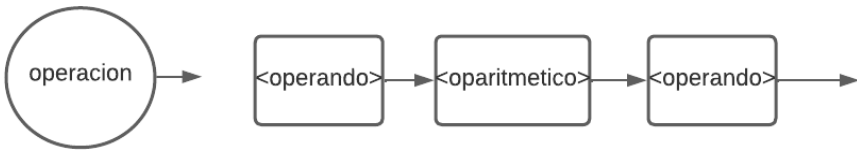
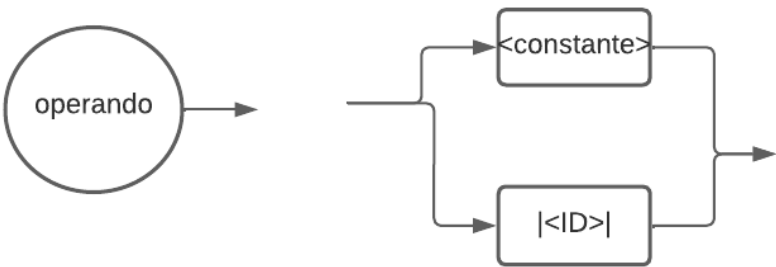
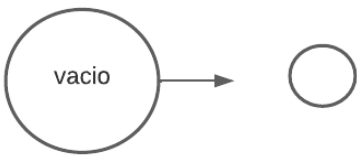
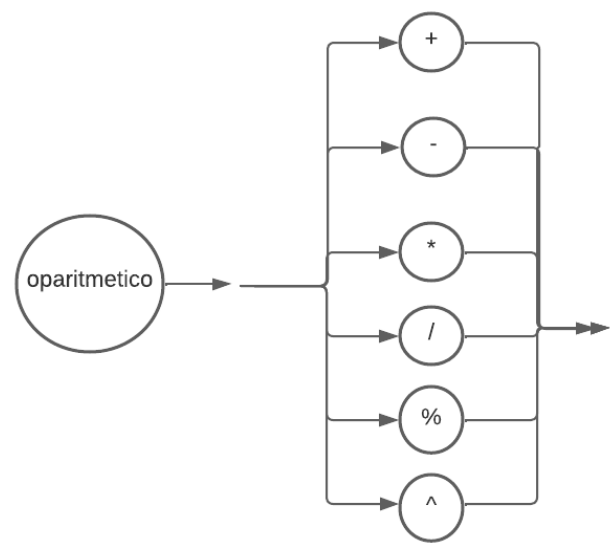
<ID> :: <minuscula> | <ID> <minuscula>

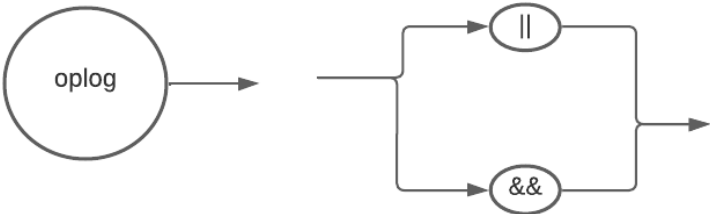
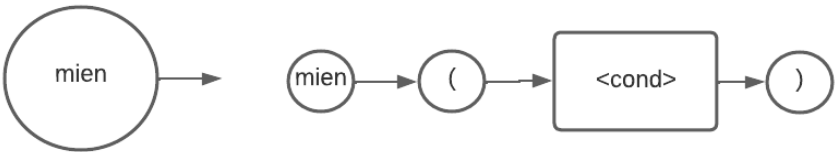
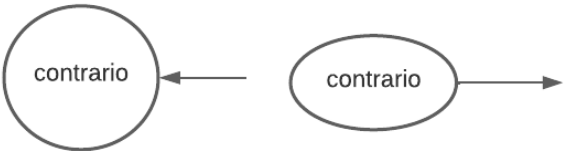
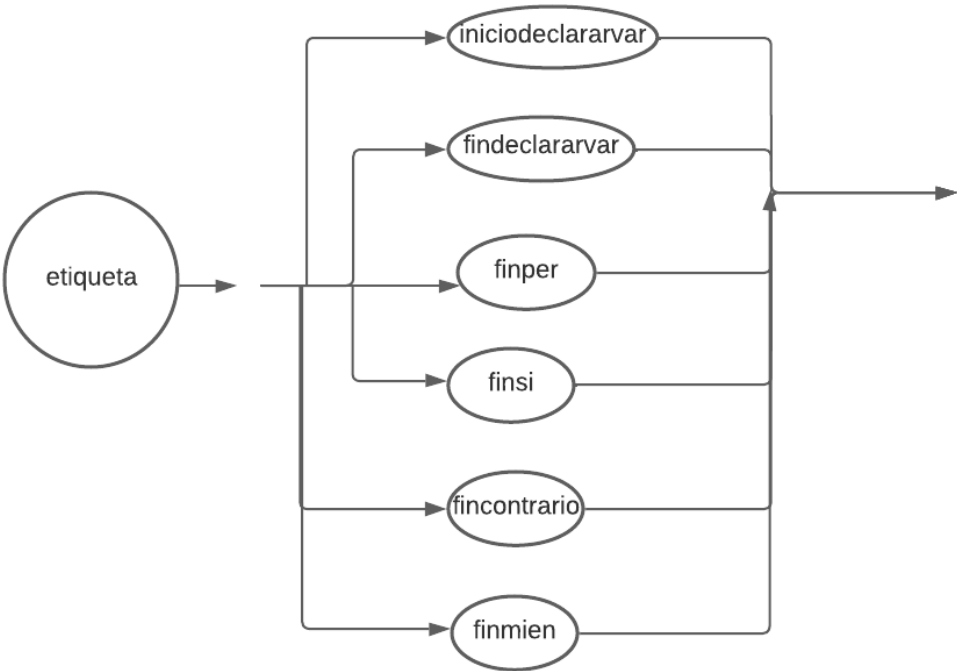
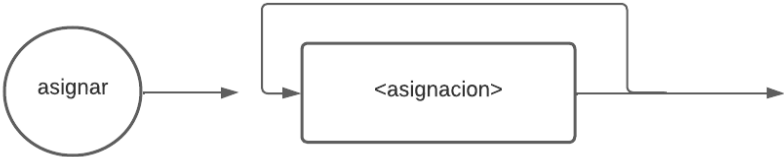
<decla> :: \ | <ID> \ | <tipodedato> ;

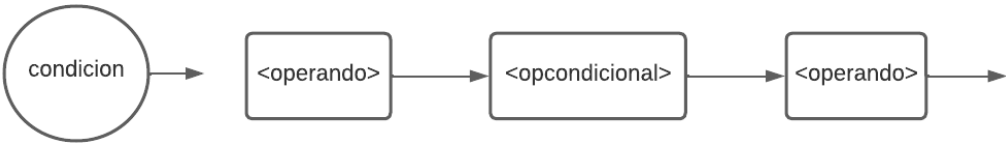
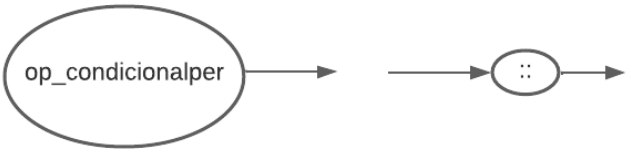
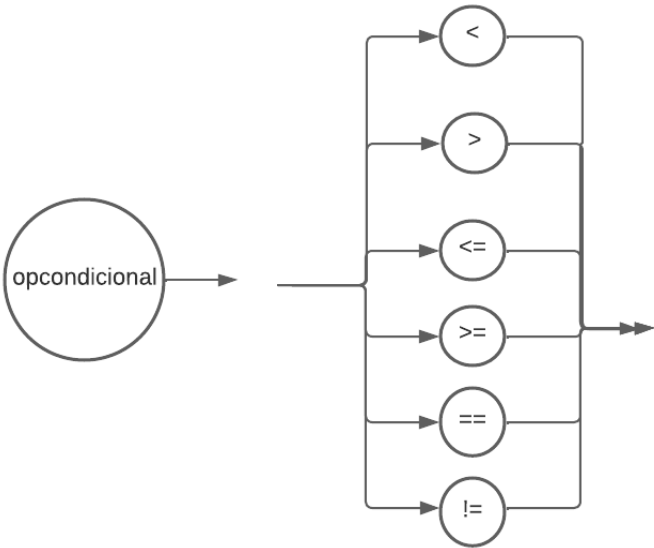
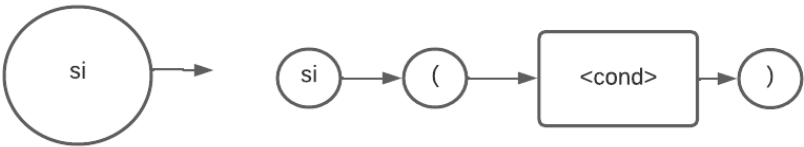
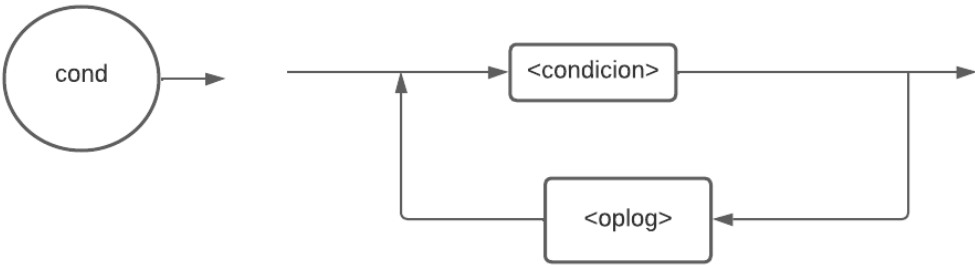
<declarar> :: <decla> | <declarar> <decla>

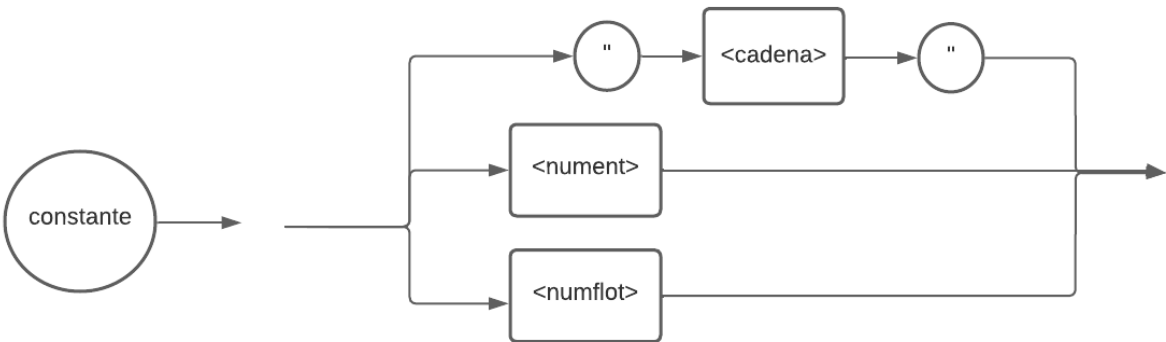
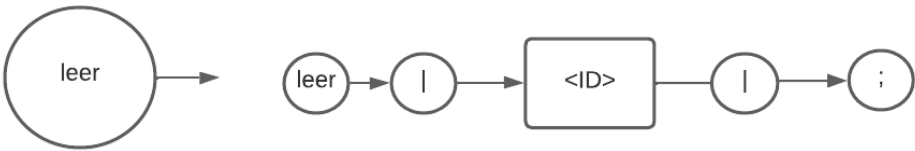
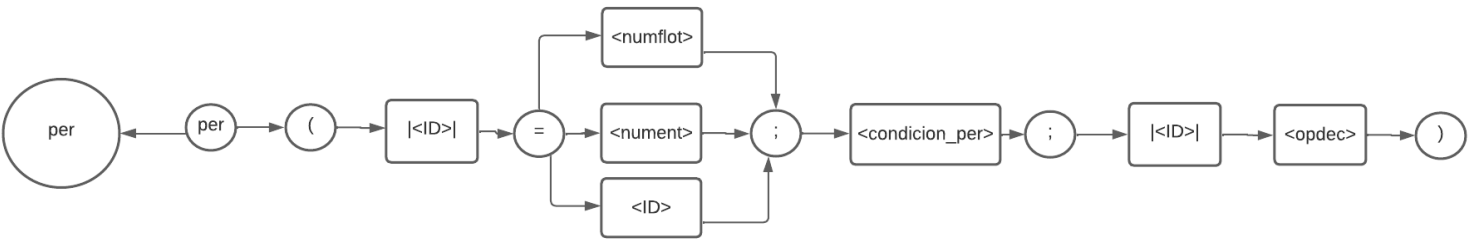
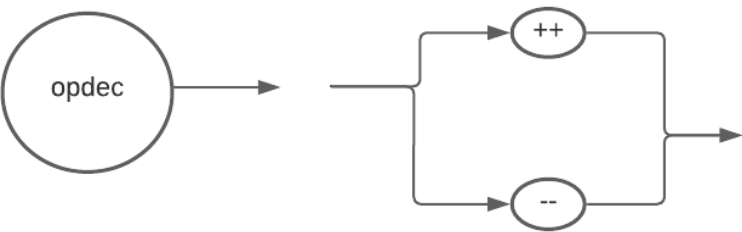
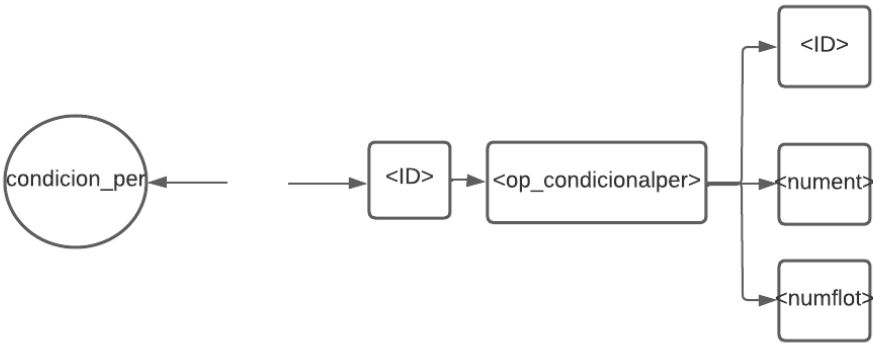
<declaracion> :: <declarar> | <vacio>

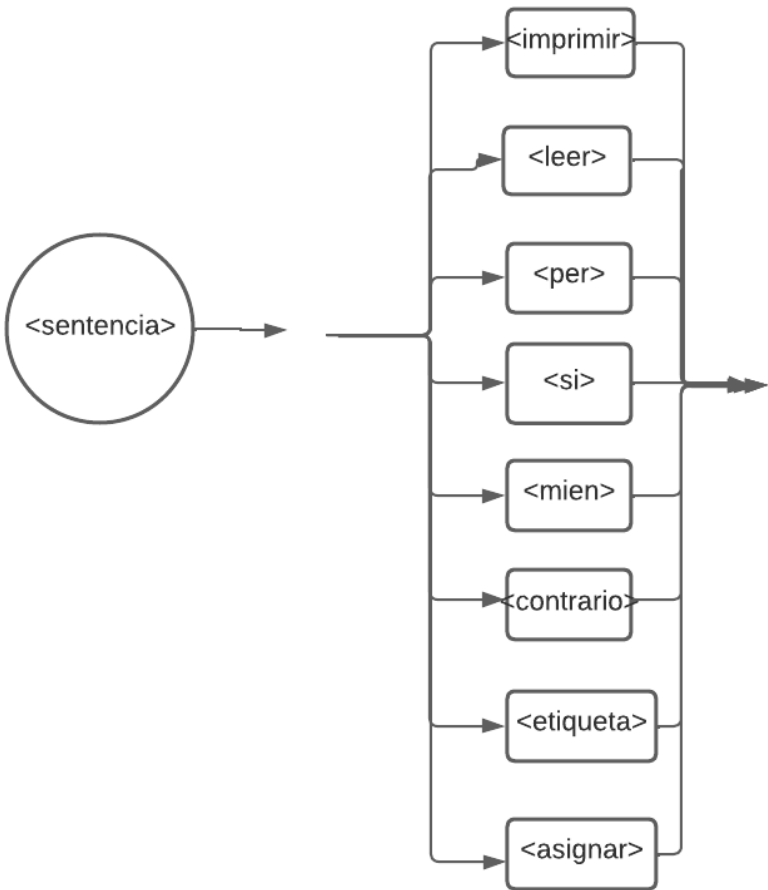
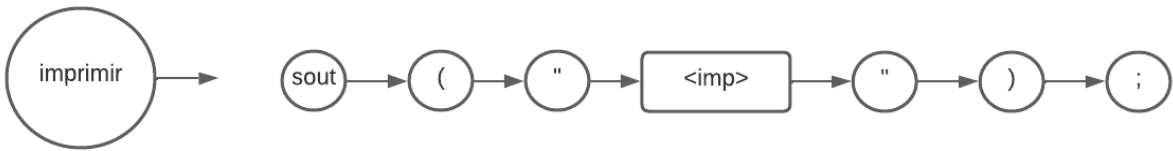
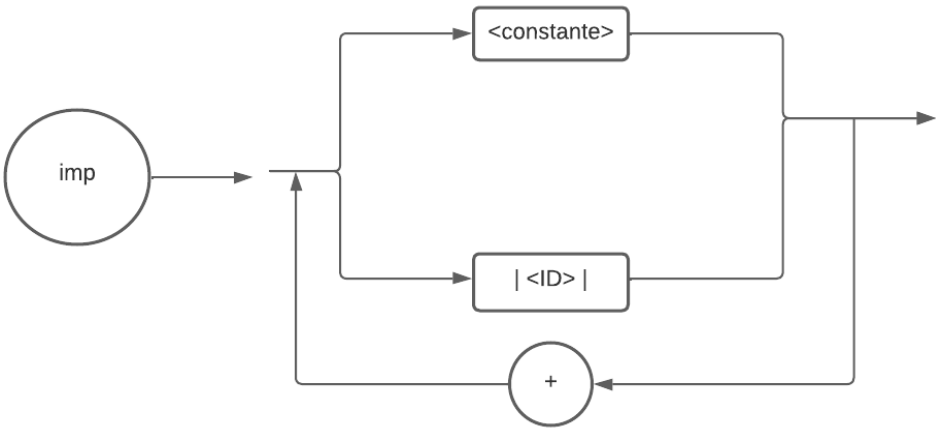
<programa> :: iniciodeclararvar <declaracion> findeclararvar | <sentencias>

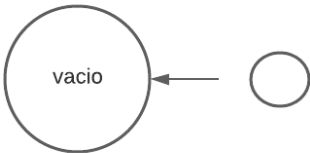
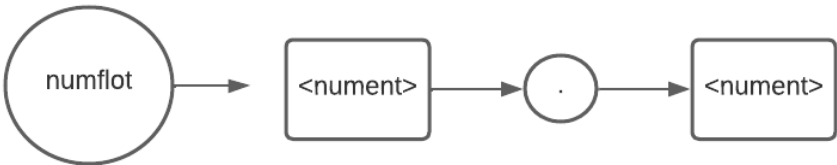
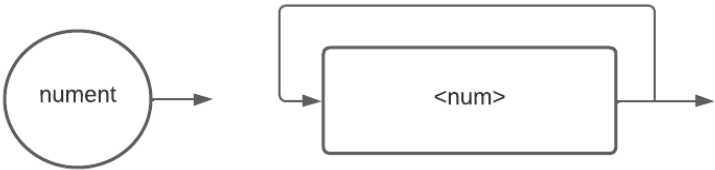
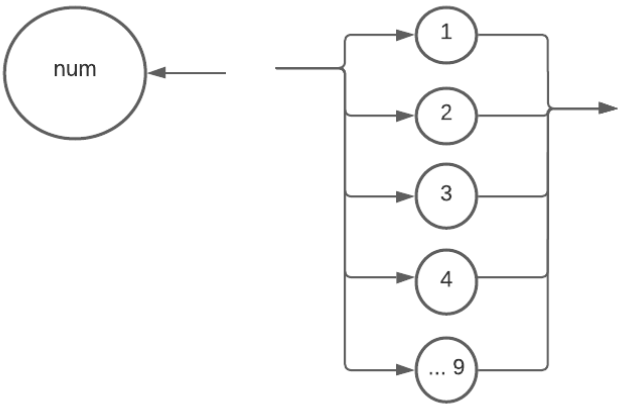
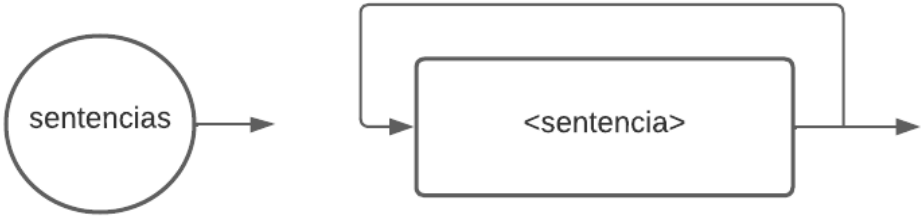




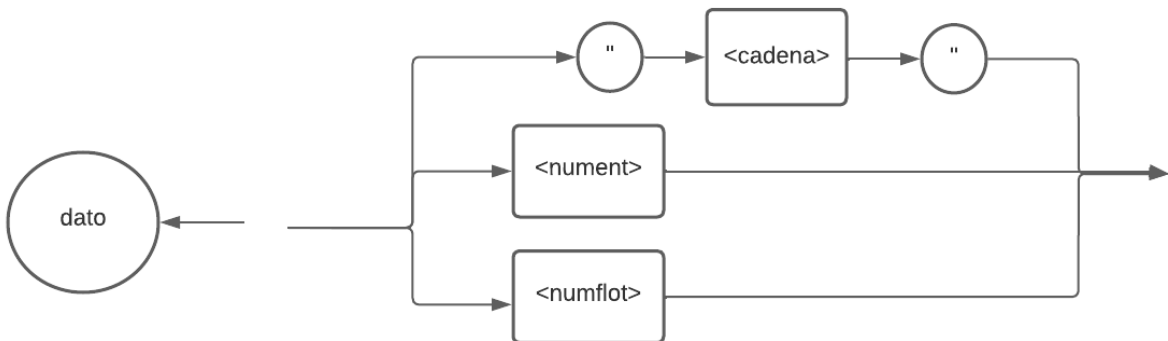
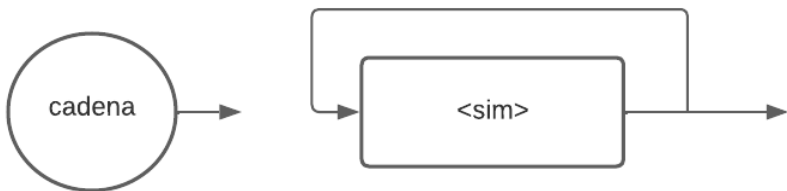
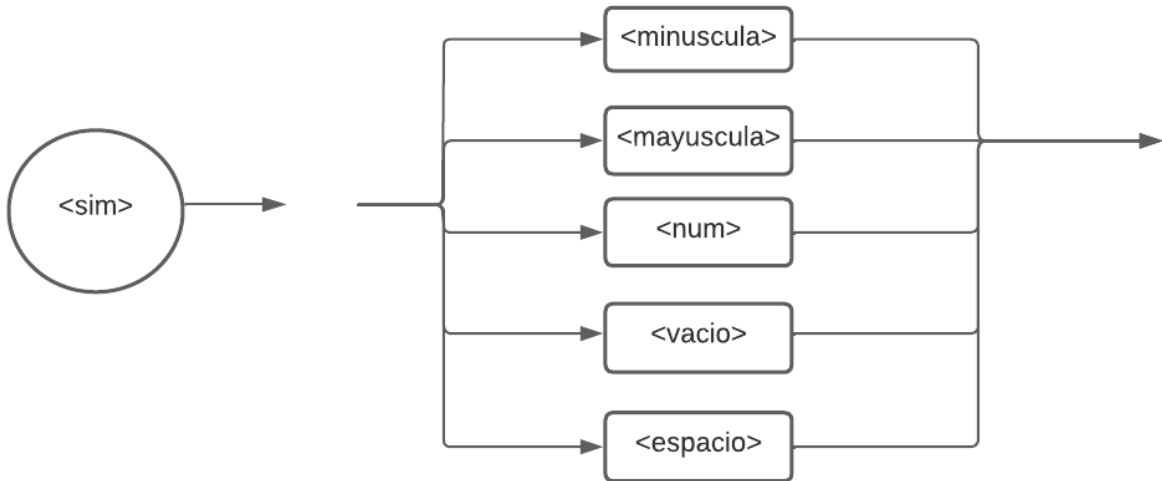
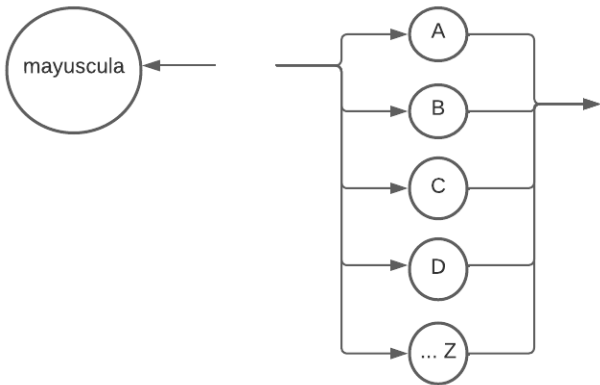


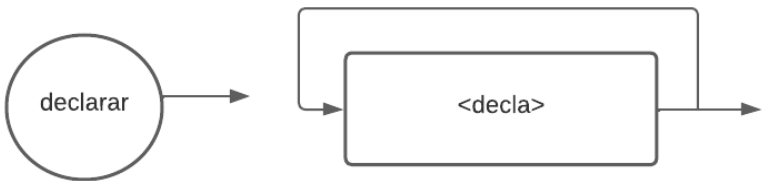
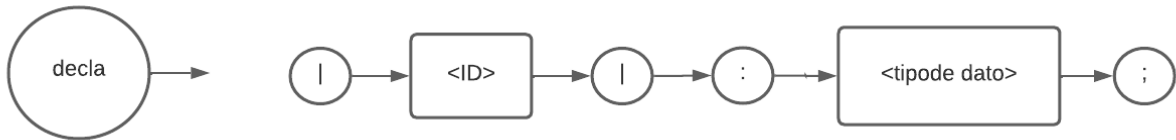
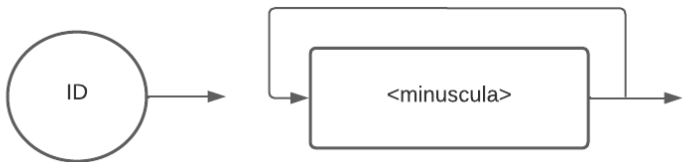
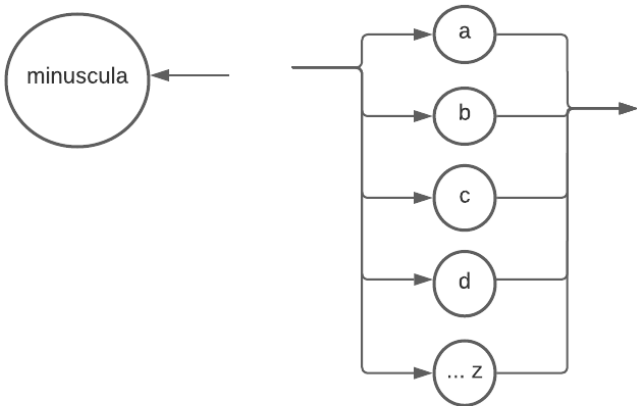
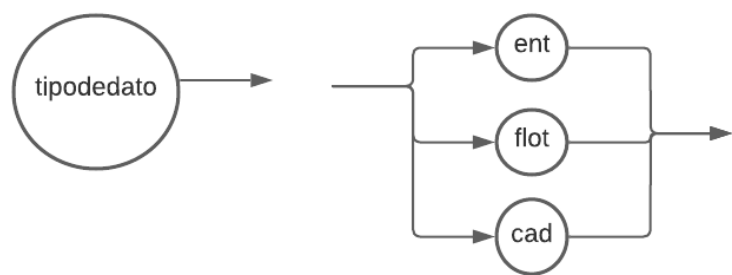


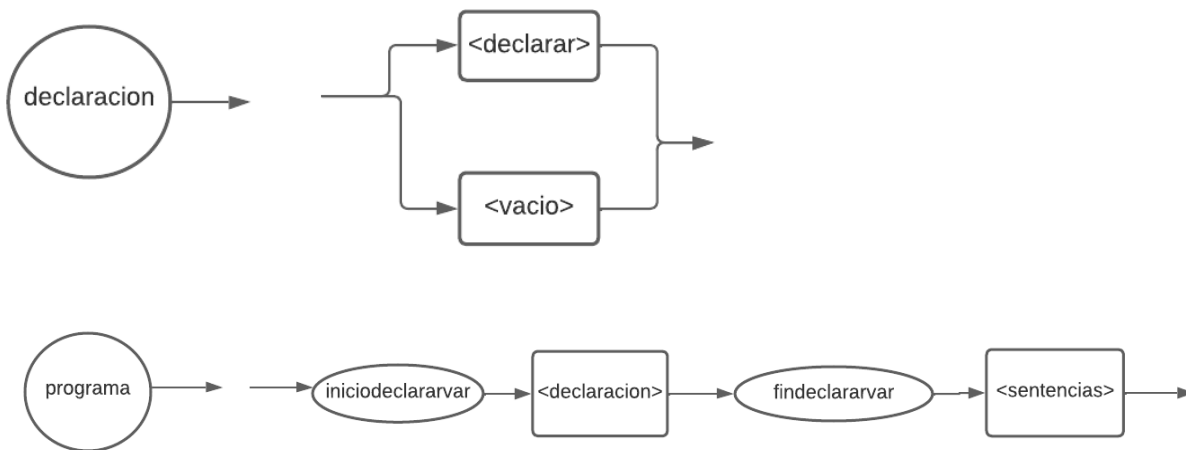












¿Qué es la pila semántica y cómo funciona?

¿En qué acciones de mi compilador interviene la pila semántica?

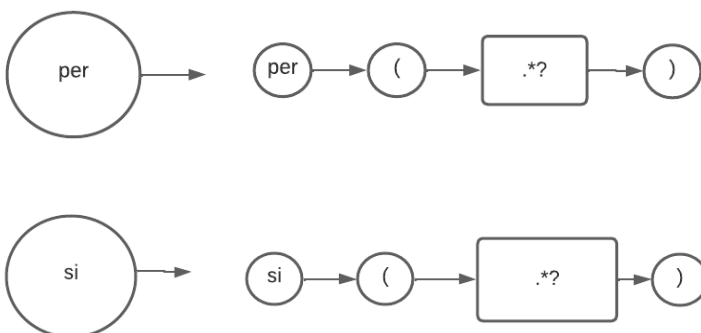
Describir cómo lo haría para cada caso.

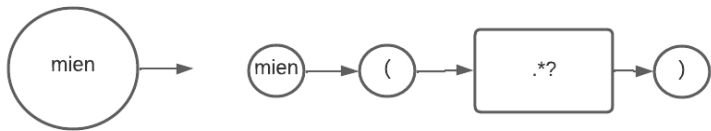
1. Lista de acciones semánticas por sentencia.
  - Verificar que la variable declarada no se repita.
  - Verificar la paridad de etiquetas que abren y cierran la sentencia.
2. Listado de errores semántico.

Notas:

- No se aceptan operaciones de más de dos operandos.
- Solo se puede imprimir un operando y no se puede concatenar.
- No existen etiquetas de fin e inicio

Para meter a la pila semántica, aunque la condición no este bien escrita debe analizar la paridad de etiquetas.





Sentencia	Elementos semánticos	Acción semántica	Errores	Acciones secundarias.
Declarar. <declaracion> :: <declarar>   <vacio>	ID Dato	Verificar si mi ID ya se encuentra en mi tabla de símbolos para evitar repetirlos. Puede agregar el ID solo en caso de que se declare dentro del bloque de declaración y de que no esté repetido, en caso contrario, no puede agregar el ID a la tabla de símbolos.	<ul style="list-style-type: none"><li>Variable repetida.</li><li>Variable no declarada (cuando declaramos fuera del bloque).</li></ul>	Agregar el ID a la tabla de símbolo en caso de que no encuentre errores.
Imprimir <imprimir> :: sout (<imp>); <imp> :: <constante>   \> <ID> \>   <imp>+ \> <ID> \>   <imp>+ <constante> •	ID Constante	Buscar ID en la tabla de símbolos y verificar si esta declarado. (Hecho desde léxico)	<ul style="list-style-type: none"><li>Variable no declarada.</li></ul>	Seguir con la verificación semántica si no hay errores.
Leer. <leer> :: leer   \> <ID> \>   ;	ID	Buscar en la tabla de símbolos si el ID fue declarado. (Ya hecho).	<ul style="list-style-type: none"><li>Variable no declarada.</li></ul>	Seguir con la verificación semántica si no hay errores.
Per. <per> :: per ( \> <ID> \> = <constante>   \> ID \> ; <condicion> ; \> <ID> \> <opdec> )	ID	Buscar en la tabla de símbolos si el ID fue declarado. (Lista) Verificar que todos los operandos sean enteros.	<ul style="list-style-type: none"><li>Variable no declarada.</li><li>Tipos de datos no coinciden.</li><li>Falta etiqueta de cierre.</li></ul>	Seguir con la verificación semántica si no hay errores.



		<p>Asegurar que la sentencia per tenga su etiqueta de cierre finper.</p> <p>En caso de no tenerla o de que no corresponda la etiqueta con la sentencia, indica un error.</p> <p>Si la línea de la sentencia no está bien escrita, se verifica si tiene etiqueta de cierre pero no se realizan acciones semánticas en la condición.</p>	<ul style="list-style-type: none"><li>• Etiqueta de cierre sin una sentencia abierta.</li></ul>	
<p>Si</p> <p>&lt;si&gt; : : si (&lt;cond&gt;)</p>	<p>ID</p> <p>Constante</p>	<p>Buscar en la tabla de símbolos si el ID fue declarado.</p> <p>Asegurar que la sentencia si tenga su etiqueta de cierre finsi.</p> <p>En caso de no tenerla o de que no corresponda la etiqueta con la sentencia, indica un error.</p> <p>Si la línea de la sentencia no está bien escrita, si se verifica si tiene etiqueta de cierre pero no se realizan acciones semánticas en la condición.</p>	<ul style="list-style-type: none"><li>• Variable no declarada.</li><li>• Falta etiqueta de cierre.</li><li>• Etiqueta de cierre sin una sentencia abierta.</li></ul>	<p>Seguir con la verificación semántica si no hay errores.</p>
<p>Mien.</p> <p>&lt;mien&gt; : : mien (&lt;cond&gt;)</p>	<p>ID</p> <p>Constante</p>	<p>Buscar en la tabla de símbolos si el ID fue declarado.</p> <p>Asegurar que la sentencia mien tenga su etiqueta de cierre finmien.</p>	<ul style="list-style-type: none"><li>• Variable no declarada.</li><li>• Falta etiqueta de cierre.</li><li>• Etiqueta de cierre sin una</li></ul>	<p>Seguir con la verificación semántica si no hay errores.</p>



		<p>En caso de no tenerla o de que no corresponda la etiqueta con la sentencia, indica un error.</p> <p>Si la línea de la sentencia no está bien escrita, si se verifica si tiene etiqueta de cierre pero no se realizan acciones semánticas en la condición.</p>	sentencia abierta.	
<p>Asignación.</p> <p>&lt;asignacion&gt; : : \   &lt;ID&gt; \   = \   &lt;ID&gt; \   \   &lt;ID&gt; \   = &lt;constante&gt;   \   &lt;ID&gt; \   = &lt;operacion&gt; ;</p> <p>&lt;operacion&gt; : : &lt;operando&gt; &lt;oparitmetico&gt; &lt;operando&gt;</p>	ID Constante	<p>Buscar en la tabla de símbolos si el ID fue declarado.</p> <p>Verificar que el operando (ID o constante) corresponda al tipo de dato del ID al que le queremos asignar el valor.</p> <p>Y asegurarnos que la operación sea correspondiente a dos operandos del mismo tipo de dato, es decir, no se puede sumar un entero con un flotante.</p> <p>A excepción de la concatenación de cadenas.</p>	<ul style="list-style-type: none"><li>• Variable no declarada.</li><li>• Tipos de datos no coinciden.</li><li>• Operación inválida.</li></ul>	Seguir con la verificación semántica si no hay errores.