

EJEMPLO DE UNA GRAMÁTICA LIBRE DEL CONTEXTO Y EL ANALIZADOR SINTÁCTICO QUE REPRESENTA

GRAMÁTICA LIBRE DEL CONTEXTO, para reconocer la asignación de una o más variables, por ejemplo:

```
N1=5; N2=40;  
C1="Hola";
```

Terminales = { tk_id, tk_igual, tk_numero, tk_cadena, tk_ptoComa }

No terminales = { <INICIO>, <ASIGNACION>, <OTRA ASIGNACION>, <VALOR> }

Símbolo Inicial = <INICIO>

Producciones:

<INICIO> -> <ASIGNACION> <OTRA ASIGNACION>

<ASIGNACION> -> tk_id tk_igual <VALOR> tk_PtoComa

<OTRA ASIGNACION> -> <ASIGNACION> <OTRA ASIGNACION>

| épsilon

<VALOR> -> tk_numero | tk_cadena

DESARROLLO DE LA GRAMÁTICA ANTERIOR (Parser), cada No terminal es una función o procedimiento; y cada terminal es la validación del token obtenido por el scanner.

Por ejemplo, para el No terminal <ASIGNACION>

Procedimiento Asignacion ()

```
{ Tk int;  
  Cad1, Cad2 string;  
  Tk=0;  
  Tk=scanner.ObtenerToken(f, c);  
  Cad1=scanner.ObtenerLexema();  
  if (Tk==tk_id) //Valida el token obtenido desde el scanner  
  { Tk=scanner.ObtenerToken(f, c);  
    if (Tk==tk_igual)  
    { Cad2=Valor(); //Llamada a una función (No terminal <VALOR>)  
      Tk=scanner.ObtenerToken(f, c);  
      if (Tk==tk_PtoComa)  
      { //Cad1=id de variable y Cad2=valor  
        GuardarAsignacion(Cad1, Cad2);  
      } else  
      { EscribirMensaje ("Se esperaba símbolo ;, fila: ", f, " col: ", c);  
      }  
    } else  
    { EscribirMensaje ("Se esperaba símbolo =, fila: ", f, " col: ", c);  
    }  
  } else  
  { EscribirMensaje ("Se esperaba identificador de variable, fila: ", f, " col: ", c);  
  }  
}
```