

# **Sintaxis y Semántica de los Lenguajes**

Facultad Regional Delta,  
Universidad Tecnológica Nacional

## **Trabajo de Laboratorio N°1**

(29 de Abril de 2019)

### **Objetivo**

Implementar un analizador lexicográfico para una gramática especificada.

### **Enunciado**

La implementación del analizador lexicográfico se realizará en grupos de 3 integrantes como máximo. Cada grupo recibirá una gramática para implementar el analizador. Se pretende desarrollar un AF por cada tipo de token y luego el analizador deberá ser implementado mediante un AFD que incluye a todos los AF construidos para cada token.

El programa que resulte de la implementación deberá aceptar una cadena que representa código escrito en el lenguaje generado por la gramática provista. Este código, visto como una cadena de caracteres ASCII, deberá ser convertido a una cadena de tokens correspondiente a la gramática provista.

### **Entrega de informe**

Deberá ser entregado un informe donde figure:

- Carátula (Universidad, Regional, Carrera, Materia, integrantes del grupo, docentes, año)
- Enunciado con la gramática que se les asignó.
- Explicación, observaciones y comentarios sobre el trabajo realizado (Por ejemplo: como se implementó el analizador, explicación de las ideas usadas en la implementación del algoritmo, estructuras de datos utilizadas, cuáles y porque fueron elegidas, más otras explicaciones que crean convenientes para entender su trabajo).
- Ejemplos de cadenas de prueba con los resultados pertinentes.

### **Gramática**

- El simbolo distinguido es Programa.
- Los terminales se hallan entre “ ”, por ejemplo “var” es un terminal llamado ***var***.
- Los no terminales **no** se hallan entre “ ” y siempre comienzan en mayúsculas, pudiendo contener mayúsculas intermedias para aclarar su significado.
- Terminales y No Terminales se hallan separados por espacios en blanco para claridad de la gramática.

## Gramática Sintáctica

Programa → ListaDecl "eof"  
ListaDecl → ListaDecl Declaracion | λ  
Declaracion → FunDecl | VarDecl | Sentencia  
FunDecl → "fun" Funcion  
Funcion → Identificador "(" ListaParametros ")" Bloque  
ListaParametros → λ | Parametros  
Parametros → Identificador | Parametros "," Identificador  
VarDecl → "var" Identificador ";" | "var" Identificador "=" Expresion ";"  
Sentencia → ExprSent | ForSent | IfSent | ReturnSent | WhileSent | Bloque  
ExprSent → Expresion ";"  
Expresion → Asignacion  
Asignacion → Identificador "=" Primitivo | OLogico;  
ForSent → "for" "(" PriArg AdicArg ";" AdicArg ")" Sentencia  
PriArg → VarDecl | ExprSent | ";"  
AdicArg → λ | Expresion  
IfSent → "if" "(" Expresion ")" Sentencia "else" Sentencia |  
          "if" "(" Expresion ")" Sentencia  
ReturnSent → "return" Expresion ";" | "return" ";"  
WhileSent → "while" "(" Expresion ")" Sentencia  
Bloque → "{" ListaSent "}"  
ListaSent → Sentencia ListaSent | λ  
OLogico → YLogico | YLogico "or" OLogico  
YLogico → Igua | Igua "and" YLogico  
Igua → Comparacion | Comparacion "==" Igua | Comparacion "!=" Igua  
Comparacion → Suma | Suma ">" Comparacion | Suma ">=" Comparacion |  
              Suma "<" Comparacion | Suma "<=" Comparacion  
Suma → Mult | "-" Suma | "+" Suma  
Mult → Unario | "/" Mult | "\*" Mult  
Unario → "!" Unario | "-" Unario | Primitivo  
Primitivo → "true" | "false" | Numero | String | Identificador |  
           "("Expresion ")"

## Gramática Léxica

Numero → ListaDigito | ListaDigito "." ListaDigito  
ListaDigito → Digito | Digito ListaDigito  
String → "" ListaSimbolos ""  
Identificador → Letra | Letra ListaSimbolos  
ListaSimbolos → Letra | Digito | Letra ListaSimbolos | Digito ListaSimbolos  
Letra → "a" ... "z" | "A" ... "Z"  
Digito → "0" ... "9" ;