

# REGLAS GRAMÁTICALES ASIGNADAS

Santiago Zerpa

Leg. 52168}

grammar Lenguaje;

programa: (declaracion | funcion | ejecucion)\* EOF;

declaracion: VAR nombre (IGUAL valor)? PYC;

funcion: FUNCION nombre (LPAREN argumentos? RPAREN) LBRACE  
instrucciones RBRACE;

argumentos: variable (COMA variable)\*;

instrucciones: (operacion\_texto | impresion | retorno);

operacion\_texto: variable IGUAL transformacion LPAREN cadena  
RPAREN PYC;

concatenar: variable IGUAL cadena MAS cadena PYC;

transformacion: MAYUSCULAS | MINUSCULAS | LONGITUD | INVERTIR |  
REEMPLAZAR;

impresion: IMPRIMIR LPAREN valor RPAREN PYC;

retorno: DEVOLVER valor PYC;

ejecucion: impresion | retorno | operacion\_texto | concatenar;

valor: texto | numero | variable;

cadena: texto | variable;

texto: CADENA;

numero: NUMERO;

variable: IDENTIFICADOR;

nombre: IDENTIFICADOR;

// Lexemas

VAR: 'variable';

FUNCION: 'funcion';

DEVOLVER: 'devolver';

IMPRIMIR: 'imprimir';

MAYUSCULAS: 'mayusculas';

MINUSCULAS: 'minusculas';

```
LONGITUD: 'longitud';
INVERTIR: 'invertir';
REEMPLAZAR: 'reemplazar';

IGUAL: '=';
MAS: '+';
COMA: ',';
PYC: ';';
LPAREN: '(';
RPAREN: ')';
LBRACE: '{';
RBRACE: '}';
EXCLAMACION: '!';

CADENA: '"' (~["\\] | '\\\' .)* '"' ;
NUMERO: [0-9]+;
IDENTIFICADOR: [a-zA-Z_][a-zA-Z0-9_]*;
NEWLINE: '\\r'? '\\n' -> skip;
WS: [ \\t\\r\\n]+ -> skip;
```