



**DEPARTAMENTO  
DE COMPUTACION**

Facultad de Ciencias Exactas y Naturales - UBA

# Trabajo Práctico Número 1

---

6 de julio de 2016

Teoría de Lenguajes

**Grupo n\_n**

Integrante	LU	Correo electrónico
Gasco, Emilio		gascoe@gmail.com
Gatti, Mathias	477/14	mathigatti@gmail.com
Patané, Federico		fede_river_8e@hotmail.com



**Facultad de Ciencias Exactas y Naturales**  
**Universidad de Buenos Aires**

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (54 11) 4576-3359

<http://www.fcen.uba.ar>

# Índice

<b>0. Introducción</b>	<b>3</b>
<b>1. La Gramática</b>	<b>4</b>
<b>2. El Código</b>	<b>9</b>
<b>3. Casos de Prueba</b>	<b>10</b>

## 0. Introducción

# 1. La Gramática

```

<programa>      :   <sentencia> <programa'>
                  |   <control> <programa'>
                  |   COMENTARIO <programa>

<programa'>     :   <sentencia> <programa'>
                  |   <control> <programa'>
                  |   COMENTARIO <programa'>
                  |   <empty>

<sentencia>     :   <var_asig> ';'
                  |   <funcion> ';'

<control>       :   <ifelse>
                  |   <loop>

<loop>          :   'while' '(' <valores> ')' <bloque>
                  |   'do' <bloque> 'while' '(' <valores> ')' ';'
                  |   'for' '(' <var_asig> ',' <valores> ',' <exp_arit> ')' <bloque>

<ifelse>        :   'if' '(' <valores> ')' <bloque>
                  |   'if' '(' <valores> ')' <bloque> else <bloque>

<bloque>        :   COMENTARIO <bloque>
                  |   <sentencia>
                  |   <control>
                  |   '{' <programa'> '}'

<funcion>       :   func_ret
                  |   func_void

<func_void>     :   'print' '(' <valores> ')

<func_ret>      :   <func_ret_int>
                  |   <func_ret_cadena>
                  |   <func_ret_bool>
                  |   <func_ret_arreglo>

<func_ret_bool> :   'colineales' '(' <valores> ',' <valores> ')

<func_ret_cadena> :   'capitalizar' '(' <valores> ')

<func_ret_arreglo> :   'multiplicarEscalar' '(' <valores> ',' <valores> ')
                  |   'multiplicarEscalar' '(' <valores> ',' <valores> ',' <valores>

<func_ret_int>  :   'length' '(' <valores> ')

```

```

<valores>      :   <exp_arit>
                  |   <comparacion>
                  |   <exp_cadena>
                  |   <exp_arreglo>
                  |   <registro>
                  |   VARIABLE
                  |   VARIABLE '[' <exp_arit> ']'
                  |   VARIABLE '[' VARIABLE ']'
                  |   VARIABLE '.' VARIABLE

<exp_arreglo>  :   '[' <list_valores> ']'
                  |   '[' ']'
                  |   func_ret_arreglo

<lista_valores> :   <valores>
                  |   <valores> ',' <lista_valores>

<registro>     :   '{' <reg_item> '}'

<reg_item>     :   CADENA ':' <valores> ',' <reg_item>
                  |   CADENA ':' <valores>

<var_asig_l>   :   VARIABLE
                  |   RES
                  |   VARIABLE '[' <exp_arit> ']'
                  |   VARIABLE '[' VARIABLE ']'
                  |   VARIABLE '.' VARIABLE

<var_asig>     :   <var_asig_l> '++'
                  |   '++' <var_asig_l>
                  |   <var_asig_l> '--'
                  |   '--' <var_asig_l>
                  |   <var_asig_l> '*' <valores>
                  |   <var_asig_l> '/' <valores>
                  |   <var_asig_l> '+' <valores>
                  |   <var_asig_l> '-' <valores>
                  |   <var_asig_l> '=' <valores>
                  |   VARIABLE '[' <exp_arit> ']' '=' <valores>
                  |   VARIABLE '[' VARIABLE ']' '=' <valores>
                  |   VARIABLE '.' VARIABLE '=' <valores>
                  |   <var_asig_l> '=' <op_ternario>

<op_ternario>  :   <exp_bool> '?' <exp_bool> ':' <exp_bool>
                  |   <exp_bool> '?' <exp_arit> ':' <exp_arit>

```

```

|    <exp_bool> '?' <exp_cadena> ':' <exp_cadena>

<var_oper>      :    VARIABLE '[' <exp_arit> ']'
|                VARIABLE '[' VARIABLE ']'
|                VARIABLE '.' VARIABLE

<exp_arit>       :    <exp_arit> '+' <term>
|                <exp_arit> '+' VARIABLE
|                <exp_arit> '+' <var_oper>
|                VARIABLE '+' <term>
|                <var_oper> '+' <term>
|                VARIABLE '+' VARIABLE
|                <var_oper> '+' <var_oper>
|                <exp_arit> '-' <term>
|                <exp_arit> '-' VARIABLE
|                <exp_arit> '-' <var_oper>
|                VARIABLE '-' <term>
|                <var_oper> '-' <term>
|                VARIABLE '-' VARIABLE
|                VARIABLE '-' <var_oper>
|                <term>

<arit_oper_2>    :    '*'
|                    '/'
|                    '%'

<term>           :    <term> <arti_oper_2> <factor>
|                VARIABLE <arit_oper_2> <factor>
|                <var_oper> <arit_oper_2> <factor>
|                <term> <arit_oper_2> VARIABLE
|                <term> <arit_oper_2> <var_oper>
|                VARIABLE <arit_oper_2> VARIABLE
|                <var_oper> <arit_oper_2> <var_oper>
|                <factor>

<factor>         :    <base> '^' <sigexp>
|                <var_oper> '^' <sigexp>
|                VARIABLE '^' <sigexp>
|                '-' <base>
|                <base> '++'
|                <base> '--'
|                '++' <base>
|                '--' <base>

```

		VARIABLE '++'
		'++' VARIABLE
		VARIABLE '--'
		'--' VARIABLE
		<var_oper> '++'
		'++' <var_oper>
		<var_oper> '--'
		'--' <var_oper>
		<base>
<base>	:	'(' <exp_arit> ')'
		NUMBER
		<func_int>
<sigexp>	:	'-' <exp>
		<exp>
<exp>	:	VARIABLE
		NUMBER
		'(' <exp_arit> ')'
<exp_cadena>	:	<exp_cadena> '+' <term_cadena>
		VARIABLE '+' <term_cadena>
		<term_cadena> '+' VARIABLE
		<term_cadena>
<term_cadena>	:	CADENA
		'capitalizar' '(' <valores> ')'
		'(' <exp_cadena> ')'
<comparacion>	:	<exp_cadena> <op_comp> <exp_cadena>
		VARIABLE <op_comp> <exp_cadena>
		<var_oper> <op_comp> <exp_cadena>
		<exp_cadena> <op_comp> VARIABLE
		<exp_cadena> <op_comp> <var_oper>
		<exp_arit> <op_comp> <exp_arit>
		VARIABLE <op_comp> <exp_arit>
		<var_oper> <op_comp> <exp_arit>
		<exp_arit> <op_comp> VARIABLE
		<exp_arit> <op_comp> <var_oper>
		VARIABLE <op_comp> VARIABLE
		<var_oper> <op_comp> <var_oper>
		<exp_bool> '==' <exp_bool>
		<exp_bool> '!=' <exp_bool>
		<exp_bool>

```
<exp_bool>      :   <exp_bool> AND <term_bool>
                  |   VARIABLE AND <term_bool>
                  |   <var_oper> AND <term_bool>
                  |   <exp_bool> AND VARIABLE
                  |   <exp_bool> AND <var_oper>
                  |   VARIABLE AND VARIABLE
                  |   <var_oper> AND <var_oper>
                  |   <term_bool>

<term_bool>      :   <term_bool> OR <factor_bool>
                  |   VARIABLE OR <factor_bool>
                  |   <var_oper> OR <factor_bool>
                  |   <term_bool> OR VARIABLE
                  |   <term_bool> OR <var_oper>
                  |   VARIABLE OR VARIABLE
                  |   <var_oper> OR <var_oper>
                  |   'not' factor_bool

<factor_bool>    :   BOOL
                  |   '(' <exp_bool> ')'
                  |   <func_bool>
                  |   <comparacion>

<op_comp>        :   '=='
                  |   '>'
                  |   '<'
                  |   '!='
```



## 2. El Código

### 3. Casos de Prueba