

PARSER-PY (FASE 2)

ANÁLISIS LÉXICO Y SINTÁCTICO

GRAMÁTICA

DECLARACIÓN DE VARIABLES

DECLARACION DE VARIABLES

<Declaracion> ::= <DeclaracionVariable>

| <DeclaracionArreglo>

| <DeclaracionDiccionario>

<DeclaracionVariable> ::= <Identificador> "=" <Cadena>

| <Identificador> "=" <Decimal>

| <Identificador> "=" <Entero>

| <Identificador> "=" "True"

| <Identificador> "=" "False"

<DeclaracionArreglo> ::= <Identificador> "=" "[" <ListaValores> "]"

<ListaValores> ::= <Valor>

| <Valor> "," <ListaValores>

<Valor> ::= <Cadena>

| <Entero>

| <Decimal>

| "True"

| "False"

<DeclaracionDiccionario> ::= <Identificador> "=" "{" <ListaPares> "}"

<ListaPares> ::= <EntradaDicc>

| <EntradaDicc> ";" <ListaPares>

<EntradaDicc> ::= <Identificador> ":" <Valor>

<Identificador> ::= IDENTIFICADOR

3

<Cadena> ::= CADENA

<Entero> ::= ENTERO

<Decimal> ::= DECIMAL

PARSER-PY (FASE 2)

ANÁLISIS LÉXICO Y SINTÁCTICO

GRAMÁTICA

DECLARACIÓN DE VARIABLES

DECLARACION DE VARIABLES

<Declaracion> ::= <DeclaracionVariable>

| <DeclaracionArreglo>

| <DeclaracionDiccionario>

<DeclaracionVariable> ::= <Identificador> "=" <Cadena>

| <Identificador> "=" <Decimal>

| <Identificador> "=" <Entero>

| <Identificador> "=" "True"

| <Identificador> "=" "False"

<DeclaracionArreglo> ::= <Identificador> "=" "[" <ListaValores> "]"

<ListaValores> ::= <Valor>

| <Valor> "," <ListaValores>

<Valor> ::= <Cadena>

| <Entero>

| <Decimal>

| "True"

| "False"

<DeclaracionDiccionario> ::= <Identificador> "=" "{" <ListaPares> "}"

<ListaPares> ::= <EntradaDicc>

| <EntradaDicc> "," <ListaPares>

<EntradaDicc> ::= <Identificador> ":" <Valor>

<Identificador> ::= IDENTIFICADOR

3

<Cadena> ::= CADENA

<Entero> ::= ENTERO

<Decimal> ::= DECIMAL

PARSER-PY (FASE 2)

ANÁLISIS LÉXICO Y SINTÁCTICO

ASIGNACIÓN

PRODUCCIONES

<Asignacion> ::= <AsignacionSimple> | <AsignacionMultiple>

<AsignacionSimple> ::= <Identificador> <OperadorAsignacion>

CONDICIONAL

<Expresion>

<AsignacionMultiple> ::= <ListaIdentificadores>

<OperadorAsignacion> <ListaExpresiones>

<ListaIdentificadores> ::= <Identificador>

| <Identificador> "," <ListaIdentificadores>

<ListaExpresiones> ::= <Expresion>

| <Expresion> "," <ListaExpresiones>

<Identificador> ::= IDENTIFICADOR

<Expresion> ::= <Cadena>

| <Entero>

| "True"

| "False"

<OperadorAsignacion> ::= "="

CONDICIONAL

<Condicional> ::= <If>

| <IfElse>

| <IfElif>

| <IfElifElse>

4

<If> ::= "if" <Expresion> ":" <BloqueCodigo>

<IfElse> ::= "if" <Expresion> ":" <BloqueCodigo> "else" ":"
<BloqueCodigo>

<IfElif> ::= "if" <Expresion> ":" <BloqueCodigo> <Elif>

<Elif> ::= "elif" <Expresion> ":" <BloqueCodigo> <Elif> | ε

La notación ϵ significa que puede haber cero o más bloques 'elif'

$\langle \text{IfElifElse} \rangle ::= \text{"if"} \langle \text{Expresion} \rangle \text{"."} \langle \text{BloqueCodigo} \rangle \langle \text{Elif} \rangle \text{"else"} \text{"."} \langle \text{BloqueCodigo} \rangle$

$\langle \text{BloqueCodigo} \rangle ::= \langle \text{Indentacion} \rangle \langle \text{Sentencias} \rangle$

$\langle \text{Indentacion} \rangle$ representa el espacio en blanco o tabulación que precede al bloque de código.

$\langle \text{Expresion} \rangle ::= \langle \text{ExpresionLogica} \rangle$

| $\langle \text{ExpresionAritmetica} \rangle$

| $\langle \text{ExpresionRelacional} \rangle$

| $\langle \text{ExpresionBooleana} \rangle$

$\langle \text{ExpresionLogica} \rangle ::= \langle \text{ExpresionLogica} \rangle \text{"and"} \langle \text{ExpresionLogica} \rangle$

| $\langle \text{ExpresionLogica} \rangle \text{"or"} \langle \text{ExpresionLogica} \rangle$

| $\text{"not"} \langle \text{ExpresionLogica} \rangle$

| $\text{"("} \langle \text{ExpresionLogica} \rangle \text{"}"$

| $\langle \text{ExpresionRelacional} \rangle$

$\langle \text{ExpresionAritmetica} \rangle ::= \langle \text{ExpresionAritmetica} \rangle \text{"+"} \langle \text{ExpresionAritmetica} \rangle$

| $\langle \text{ExpresionAritmetica} \rangle \text{"-"} \langle \text{ExpresionAritmetica} \rangle$

| $\langle \text{ExpresionAritmetica} \rangle \text{"*"} \langle \text{ExpresionAritmetica} \rangle$

| $\langle \text{ExpresionAritmetica} \rangle \text{" / " } \langle \text{ExpresionAritmetica} \rangle$

| $\text{"("} \langle \text{ExpresionAritmetica} \rangle \text{"}"$

| $\langle \text{Entero} \rangle$

5

| $\langle \text{Decimal} \rangle$

| $\langle \text{Identificadores} \rangle$

ASIGNACIÓN

<Asignacion> ::= <AsignacionSimple> | <AsignacionMultiple>

<AsignacionSimple> ::= <Identificador> <OperadorAsignacion>
<Expresion>

<AsignacionMultiple> ::= <ListalIdentificadores>
<OperadorAsignacion> <ListaExpresiones>

<ListalIdentificadores> ::= <Identificador>

| <Identificador> ";" <ListalIdentificadores>

<ListaExpresiones> ::= <Expresion>

| <Expresion> ";" <ListaExpresiones>

<Identificador> ::= IDENTIFICADOR

<Expresion> ::= <Cadena>

| <Entero>

| "True"

| "False"

<OperadorAsignacion> ::= "="

OPERADOR TERNARIO

<OperadorTernario> ::= <Expresion> "?" <Expresion> ":" <Expresion>

| <Expresion> "if" <Expresion> "else" <Expresion>

<Expresion> ::= <ExpresionLogica>

| <ExpresionAritmetica>

| <ExpresionRelacional>

| <ExpresionBooleana>

<ExpresionLogica> ::= <ExpresionLogica> "and" <ExpresionLogica>

| <ExpresionLogica> "or" <ExpresionLogica>

| "not" <ExpresionLogica>

| "(" <ExpresionLogica> ")"

6

| <ExpresionRelacional>

PARSER-PY (FASE 2)

ANÁLISIS LÉXICO Y SINTÁCTICO

CICLOS FOR

<CicloFor> ::= "for" <Identificador> "in" <Iterable> ":" <BloqueCodigo>

<Iterable> ::= <Rango>

| <Lista>

| <Diccionario>

| <LlamadaFuncion>

<Rango> ::= "range" "(" <Expresion> ")"

<Lista> ::= "[" <ListaValores> "]"

<Diccionario> ::= "{" <ListaPares> "}"

<LlamadaFuncion> ::= <Identificador> "(" <Argumentos> ")"

<ListaValores> ::= <Valor> | <Valor> "," <ListaValores>

<Valor> ::= <Entero>

| <Decimal>

| <Cadena>

| <Booleano>

| <Arreglo>

| <Diccionario>

<ListaPares> ::= <EntradaDicc> | <EntradaDicc> "," <ListaPares>

<EntradaDicc> ::= <Identificador> ":" <Valor>

<Expresion> ::= <ExpresionLogica>

| <ExpresionAritmetica>

| <ExpresionRelacional>

7

| <ExpresionBooleana>

| <ExpresionPertenencia>

<BloqueCodigo> ::= <Indentacion> <Sentencias>

<Indentacion> ::= INDENTACION

<Identificador> ::= IDENTIFICADOR

<Entero> ::= ENTERO

<Decimal> ::= DECIMAL

<Cadena> ::= CADENA

<Booleano> ::= "True" | "False"

<Arreglo> ::= "[" <ListaValores> "]"

PARSER-PY (FASE 2)

ANÁLISIS LÉXICO Y SINTÁCTICO

CICLOS FOR ELSE

<CicloFor> ::= "for" <Identificador> "in" <Iterable> ":" <BloqueCodigo>

<CicloForConElse> <Identificador> "in" <Iterable> ":"
<BloqueCodigo> "else" ":" <BloqueCodigo>

<Iterable> ::= <Rango>

| <Lista>

| <Diccionario>

| <LlamadaFuncion>

<Rango> ::= "range" "(" <Expresion> ")"

<Lista> ::= "[" <ListaValores> "]"

<Diccionario> ::= "{" <ListaPares> "}"

<LlamadaFuncion> ::= <Identificador> "(" <Argumentos> ")"

<ListaValores> ::= <Valor> | <Valor> "," <ListaValores>

<Valor> ::= <Entero>

| <Decimal>

| <Cadena>

| <Booleano>

| <Arreglo>

| <Diccionario>

<ListaPares> ::= <EntradaDicc> | <EntradaDicc> "," <ListaPares>

<EntradaDicc> ::= <Identificador> ":" <Valor>

<Expresion> ::= <ExpresionLogica>

| <ExpresionAritmetica>

| <ExpresionRelacional>

| <ExpresionBooleana>

CICLOS WHILE

| <ExpresionPertenencia>

<BloqueCodigo> ::= <Indentacion> <Sentencias>

<Indentacion> ::= INDENTACION

<Identificador> ::= IDENTIFICADOR

<Entero> ::= ENTERO

<Decimal> ::= DECIMAL

<Cadena> ::= CADENA

<Booleano> ::= "True" | "False"

<Arreglo> ::= "[" <ListaValores> "]"

<CicloWhile> ::= "while" <Expresion> ":" <BloqueCodigo>

<Expresion> ::= <ExpresionLogica>

| <ExpresionAritmetica>

| <ExpresionRelacional>

| <ExpresionBooleana>

| <ExpresionPertenencia>

<BloqueCodigo> ::= <Indentacion> <Sentencias>

<Indentacion> ::= INDENTACION

<ExpresionLogica> ::= <ExpresionLogica> "and" <ExpresionLogica>

| <ExpresionLogica> "or" <ExpresionLogica>

| "not" <ExpresionLogica>

| "(" <ExpresionLogica> ")"

| <ExpresionRelacional>

<ExpresionAritmetica> ::= <ExpresionAritmetica> "+"

<ExpresionAritmetica>

| <ExpresionAritmetica> "-" <ExpresionAritmetica>

| <ExpresionAritmetica> "*" <ExpresionAritmetica>

| <ExpresionAritmetica> "/" <ExpresionAritmetica>

| "(" <ExpresionAritmetica> ")"

| <Entero>

| <Decimal>

| <Identificador>

<ExpresionRelacional> ::= <ExpresionAritmetica> "=="
<ExpresionAritmetica>

| <ExpresionAritmetica> "!=" <ExpresionAritmetica>

10

| <ExpresionAritmetica> ">" <ExpresionAritmetica>

| <ExpresionAritmetica> "<" <ExpresionAritmetica>

| <ExpresionAritmetica> ">=" <ExpresionAritmetica>

| <ExpresionAritmetica> "<=" <ExpresionAritmetica>

<ExpresionBooleana> ::= "True" | "False"

<ExpresionPertenencia> ::= <Expresion> "in" <Expresion>

<Identificador> ::= IDENTIFICADOR

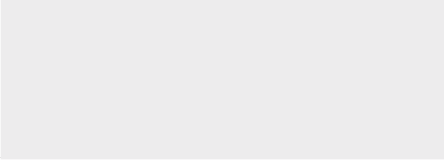
<Entero> ::= ENTERO

<Decimal> ::= DECIMAL

FUNCIÓN| MÉTODO

<Funcion> ::= "def" <Identificador> "(" <Parametros> ")" ":"
<BloqueCodigo>

<Parametros> ::= <Parametro> | <Parametro> "," <Parametros>



$\langle \text{Parametro} \rangle ::= \langle \text{Identificador} \rangle$

$\langle \text{BloqueCodigo} \rangle ::= \langle \text{Indentacion} \rangle \langle \text{Sentencias} \rangle$