

IC 5701 - Compiladores e intérpretes

1

Gramática de TerraCode

Estudiantes:

Jozafath Pérez Fernández - 2023107460

Alana Calvo Bolaños - 2022040915

Mary Paz Álvarez Navarrete - 2023138604

José Gabriel Jiménez Chacón - 2021128841

Docente:

SANABRIA RODRIGUEZ AURELIO

Año:

IS 2025

Gramáticas libres de contexto

Programa ::= Instrucciones+

Instrucciones ::= Comentario | Declaracion | FuncionCreada | Funcion | Repeticion |
Condicional | Imprimir | CicloCasos | Retorno

Comentario ::= "*" Texto "*"

Declaracion ::= TipoVar Identificador "=" Expresion
TipoVar ::= "maceta" | "semilla"

FuncionCreada ::= Identificador "(" ((ValorEstandar | Identificador) ("," (ValorEstandar |
Identificador))* "?")"

Retorno ::= "cosechar" (Expresion)?

Funcion ::= "sembrar" Identificador "(" (Parametros)? ")" "{" Instrucciones+ "}" "florecer"
Parametros ::= Identificador | ("," Identificador)*

Repeticion ::= "fotosintesis" "(" Expresion ")" "{" Instrucciones+ "}"

Condicional ::= If+ Else?
If ::= "exterior" "(" Expresion ")" "{" Instrucciones+ "}"
Else ::= "interior" "{" Instrucciones+ "}"

Expresion ::= ExpresionLogica | ExpresionAritmetica | ExpresionComparacion |
ExpresionIgualdad | ExpresionNegacion | Concatenacion | LlamadaFuncion | ValorEstandar

ValorEstandar ::= Entero | Flotante | String | ValorBooleano

LlamadaFuncion ::= Largo | Entrada | StringEntero | StringFlotante | FuncionCreada

ExpresionLogica ::= OpLogico "(" ExpresionLogicaTerm ("," ExpresionLogicaTerm)+ ")"
ExpresionLogicaTerm ::= ExpresionComparacion | ExpresionIgualdad | ValorBooleano |
ExpresionNegacion | ExpresionLogica | Identificador

ExpresionComparacion ::= OpComparacion "(" (ExpresionAritmeticaTerm) ","
(ExpresionAritmeticaTerm) ")"

ExpresionIgualdad ::= "cactus" "(" (ExpresionAritmeticaTerm | ValorEstandar) (","
(ExpresionAritmeticaTerm | ValorEstandar) ")"

ExpresionAritmetica ::= OpAritmetico "(" ExpresionAritmeticaTerm (","
ExpresionAritmeticaTerm)+ ")"
ExpresionAritmeticaTerm ::= Entero | Flotante | Identificador | ExpresionAritmetica

ExpresionNegacion ::= "acaro" ExpresionLogicaTerm

Concatenacion ::= "conocarpus" "(" (String | Identificador) (," (String | Identificador))+ ")"

OpLogico ::= "injerto" | "dioica"

OpAritmetico ::= "sumatra" | "reseda" | "bambu" | "diosma" | "dalia" | "magnolia" | "conocarpus"

OpComparacion ::= "secuoya" | "bonsai" | "rosa" | "lirio"

ValorBooleano ::= "viva" | "muerta"

Imprimir ::= "podar" "(" String | Identificador (," (String | Identificador))* ")"

Largo ::= "medir" "(" String | Identificador ")"

Entrada ::= "recolectar" "(" String ")"

StringEntero ::= "madurar" "(" String | Entero | Identificador ")"

StringFlotante ::= "germinar" "(" String | Entero | Identificador ")"

CicloCasos ::= "planta" "(" Expresion ")" "{" Instrucciones "flor" Identificador '=' Expresion Casos+ CasoDefault "}"

Casos ::= "hoja_compuesta" "(" ValorEstandar ")" "{" Instrucciones+ "}"

CasoDefault ::= "hoja_simple" "(" ")" "{" Instrucciones+ "}"

String ::= "#" Texto "#"

Texto ::= [A-Z]?[a-z]?[0-9]?[ÁÉÍÓÚáéíóúññüü]*

Identificador ::= [a-zA-Z][a-zA-Z0-9]*

Entero ::= ("+"|"-")?[0-9]+"

Flotante ::= ("+"|"-")?[0-9]+"."[0-9]+

En expresiones regulares:

Signo ? = opcional

Signo + = puede ser 1 o más

Signo * = puede ser 0 o más