**GRAMÁTICA**

class\_declaration = < modifier > "class" identifier

"{" < field\_declaration > <statement> "}"

field\_declaration = variable\_declaration ";"

variable\_declaration = < modifier > (type (variable\_declarator/identifier)"

variable\_declarator = identifier "=" integer\_literal / boolean\_literal]

expression = testing\_expression

testing\_expression = (integer\_literal / identifier

( ">"

/ "<"

/ ">="

/ "<="

/ "=="

/ "!=" )

integer\_literal / identifier)

statement = variable\_declaration / if\_statement/ while\_statement

while\_statement = "while" "(" expression ")" "{" < statement > "}"

if\_statement = "if" "(" expression ")" "{" aritmetica\_expression < statement > "}"

type = type\_specifier

type\_specifier = "boolean" / "int"

modifier = "public" / "private"

integer\_literal = ( ( "1..9" < "0..9" > )

boolean\_literal= “true” / ”false”

identifier = "a..z" "1..9" //(a-z)+ (1-9)+

aritmetica\_expression = = (identifier "=" (integer\_literal

( "+"

/ "-"

/ ”/”

/ ”\*”)

integer\_literal “;”)

ANÁLISIS LEXICO Y ANALISIS SINTACTICO

En base a la gramática realizar el análisis léxico y sintáctico e ir llenando la tabla de símbolos (investigar arboles ascendentes y descendentes para el análisis sintáctico).

Necesitas la gramática y un archivo de texto para iniciar con la verificación

En base a la gramática se tiene que realizar el analizador léxico para que solo acepte los token válidos para esta gramática. Para lo cual tendrás que realizar grupos de token los cuales se les llama componentes léxicos. Tendrás que mostrar los errores léxicos que contenga el documento.

Crear la tabla de símbolos con la información requerida para realizar las siguientes fases del compilador. Recuerda que en la primera fase se agregan los identificadores a la tabla y posteriormente toda la información relacionada con ellos,

Realizar la fase de análisis sintáctico, dado el archivo de entrada mismo archivo que se utiliza en el análisis léxico, verificar la estructura, y eso se logra comparando los token (componentes léxicos identificados en la fase lexica) con la gramática dada, recuerda que la gramática es la que determina en qué posición deben de acomodarse cada uno de los token y si al ir verificando los token no coincide , entonces enviarían un mensaje de error y el programa debe de continuar con la verificación del archivo hasta terminar de leer todos los token.

Al final lo que se obtendrá es una lista de errores si es que tuvo algún error en alguna de las fases, o un mensaje de archivo correcto si es que no tuvo algún error. Además contaran con una tabla de símbolos con la información de los identificadores.