Gramática Abstracta

CATEGORIES

definicion, expresion, sentencia, tipo

NODES

programa -> definicion\* ;

defVariable:definicion -> nombre:string tipo ambito:int;

defFuncion:definicion -> nombre:string parametros:defVariable\* retorno cuerpo;

retorno -> tipo ;

cuerpo -> defVariable\* sentencia\* ;

intType:tipo -> ;

realType:tipo -> ;

charType:tipo -> ;

identType:tipo -> nombre:string;

arrayType:tipo -> dimension:literalInt tipo ;

defEstructura:definicion -> nombre:string defCampo\* ;

defCampo -> nombre:string tipo ;

while:sentencia -> condicion:expresion cierto:sentencia\* ;

ifElse:sentencia -> condicion:expresion cierto:sentencia\* falso:sentencia\* ;

return:sentencia -> expresion ;

print:sentencia -> expresion ;

println:sentencia -> expresion ;

printsp:sentencia -> expresion ;

read:sentencia -> expresion ;

asigna:sentencia -> left:expresion right:expresion;

llamadaFuncionSentencia:sentencia -> nombre:string argumentos:expresion\* ;

expresionAritmetica:expresion -> left:expresion operador:string right:expresion;

expresionBooleana:expresion -> left:expresion operador:string right:expresion;

invocacion:expresion -> nombre:string argumentos:expresion\* ;

variable:expresion -> nombre:string;

literalInt:expresion -> valor:string;

literalReal:expresion -> valor:string;

literalChar:expresion -> valor:string;

varArray:expresion -> identificacion:expresion posicion:expresion ;

cast:expresion -> tipo expresion ;

navega:expresion -> expresion nombre:string ;

conParentesis:expresion -> expresion ;

negacion:expresion -> expresion ;