GRAMÁTICA CORREGIDA

Program -> Decl'
Decl' -> Decl (Decl' | ε)
Decl -> VariableDecl | FunctionDecl
VariableDecl -> Variable;
Variable' -> Variable (, Variable' | ε)
Variable -> Type ident
Type -> Type' ([] | ε)
Type' -> int | double | bool | string | ident
FunctionDecl -> Type FunctionDecl' | void FunctionDecl'
FunctionDecl' -> ident (Formals) Stmt'
Formals -> Variable' | ε
Stmt' -> Stmt Stmt' | ε
Stmt -> WhileStmt Stmt' | printStmt Stmt'| Expr , Stmt'
WhileStmt -> while (Expr) Stmt
PrintStmt -> Print (Expre' ,);
Expre' -> Expr Expre'
Expr -> ExprM Expr'
Expr' -> && ExprM Expr' | ||ExprM Expr' | ε
ExprM -> ExprN ExprM'
ExprM' -> == ExprNExprM' | != ExprNExprM' | >=ExprNExprM' |
<= ExprNExprM' | <ExprNExprM' | > ExprN ExprM' | ε
ExprN -> ExprO ExprN'
ExprN' -> +ExprOExprN' | -ExprOExprN' | ε
ExprO .> ExprP ExprO'
ExprO' -> / ExprPExprO' | *ExprPExprO' | %ExprPExprO' | ε
ExprP -> Lvalue = Expr | Constant | Lvalue | this | (Expr) | -Expr | !Expr |new(ident)
LValue -> ident | Expr.ident | Expr [Expr]
Constat-> intConstant | doubleConstant | boolConstant | stringConstant | null

(Los símbolos subrayados con azul son símbolos terminales).

NO.	Terminales	No terminales
0	;	Program
1	ident	Decl’
2	int	Decl
3	double	VariableDecl
4	bool	Variable’
5	string	Variable
6	[]	Type
7	void	FunctionDecl
8	(	FunctionDecl’
9	)	Formals
10	ε	Stmt’
11	,	Stmt
12	while	WhileStmt
13	Print	PrintStmt
14	&&	Expre’
15	||	Expr
16	==	Expr’
17	!=	ExprM
18	>=	ExprM’
19	<=	ExprN
20	<	ExprN’
21	>	ExprO
22	+	ExprO’
23	-	ExprP
24	/	LValue
25	*	Constant
26	%	
27	=	
28	this	
29	!	
30	New	
31	intConstant	
32	doubleConstant	
33	boolConstant	
34	stringConstant	
35	null	