

Gramática con logros

Inicio -> Clase Clases \$
Clases -> Clase Clases | ϵ
Clase -> class idClase Herencia { Miembros }
Herencia -> extends idClase | ϵ
Miembros -> Miembro Miembros | ϵ
Miembro -> Atributo | Ctor | Metodo
Atributo -> Visibilidad Tipo ListaDecVars ;
Metodo -> FormaMetodo TipoMetodo idMetVar ArgsFormales Bloque
Ctor -> idClase ArgsFormales Bloque
ArgsFormales -> (ListaArg)
ListaArg -> ϵ | Arg , ArgFormales
ArgFormales -> ϵ | Arg , ArgFormales
Arg -> Tipo idMetVar
FormaMetodo -> static | dynamic
Visibilidad -> public | private
TipoMetodo -> Tipo | void
Tipo -> TipoPrimitivo | TipoReferencia
TipoPrimitivo -> boolean | char | int
TipoReferencia -> idClase | String | TipoArreglo [] /* arreglo de todo */
TipoArreglo -> boolean | char | int | idClase | String /* arreglo de todo */
ListaDecVars -> idMetVar ListaDV
ListaDV -> , idMetVar ListaDV | ϵ
Bloque -> { Sentencias }
Sentencias -> Sentencia Sentencias | ϵ
Sentencia -> ;
Sentencia -> Asignacion ;
Sentencia -> SentenciaLlamada ;
Sentencia -> Tipo ListaDecVars Inicializacion ; /* declaracion asignacion */
Sentencia -> if (Expresion) Sentencia SentenciaElse
Sentencia -> while (Expresion) Sentencia
Sentencia -> Bloque
Sentencia -> return Expresiones ;
SentenciaElse -> ϵ | else Sentencia
Inicializacion -> = Expresion | ϵ /* declaracion asignacion */
Expresiones -> Expresion Expresiones | ϵ
Asignacion -> AccesoVar = Expresion | AccesoThis = Expresion
SentenciaLlamada -> Primario /* llamada sin parentisis */
Expresion -> ExpOr
ExpOr -> ExpAnd ExpOrR
ExpOrR -> || ExpAnd ExpOrR | ϵ
ExpAnd -> ExpIg ExpAndR
ExpAndR -> && ExpIg ExpAndR | ϵ
ExpIg -> ExpComp ExpIgR
ExpIgR -> OpIgual ExpComp ExpIgR | ϵ
ExpComp -> ExpAd ExpCompR
ExpCompR -> OpComp ExpAd | ϵ
ExpAd -> ExpMul ExpAdR
ExpAdR -> OpAd ExpMul ExpAdR | ϵ
ExpMul -> ExpUn ExpMulR
ExpMulR -> OpMul ExpUn ExpMulR | ϵ

ExpUn -> OpUn ExpUn | Operando
 OpIgual -> == | !=
 OpComp -> < | > | <= | >=
 OpAd -> + | -
 OpUn -> + | - | !
 OpMul -> * | /
 Operando -> Literal
 Operando -> Primario
 Literal -> null | true | false | intLiteral | charLiteral | stringLiteral
 Primario -> ExpresionParentizada
 Primario -> AccesoThis
 Primario -> idMetVar MetodoVariable
 Primario -> LlamadaMetodoEstatico
 Primario -> LlamadaCtor
 MetodoVariable -> Encadenado | ArgsActuales Encadenado
 ExpresionParentizada -> (Expresion) Encadenado
 Encadenado -> ϵ | . idMetVar Acceso | AccesoArregloEncadenado
 Acceso -> LlamadaMetodoEncadenado | AccesoVarEncadenado
 AccesoThis -> this Encadenado
 AccesoVar -> idMetVar Encadenado
 LlamadaMetodo -> idMetVar ArgsActuales Encadenado
 LlamadaMetodoEstatico -> idClase . LlamadaMetodo Encadenado
 LlamadaCtor -> new LlamadaCtorR
 LlamadaCtorR -> idClase ArgsActuales Encadenado | TipoArreglo [Expresion]
 Encadenado /* arreglo de todo */
 ArgsActuales -> (ListaExpresiones)
 ListaExpresiones -> Expresion ListaExp | ϵ
 ListaExp -> , Expresion ListaExp | ϵ
 LlamadaMetodoEncadenado -> ArgsActuales Encadenado
 AccesoVarEncadenado -> Encadenado
 AccesoArregloEncadenado -> [Expresion] Encadenado /* arreglo de todo?*/