

Regla	Primero	siguiente
programa	programa	#
declaraciones	variables €	procedimiento funcion comienzo
lista_de_ident	ID	:
resto_lista_ident	, €	:
tipo	entero real booleano array	; variables comienzo)
lista_de_enteros	INTEGER]
resto_lista_enteros	, €]
decl_de_subprogs	procedimiento funcion €	comienzo
decl_de_procedimiento	procedimiento	procedimiento funcion comienzo
decl_de_funcion	funcion	procedimiento funcion comienzo
cabecera_procedimiento	procedimiento	variables comienzo
cabecera_funcion	funcion	variables comienzo
argumentos	(€	variables comienzo retorna
lista_de_param	ID)
resto_lis_de_param	; €)
clase_param	entrada salida	entero real booleano array
clase_param'	salida €	entero real booleano array
lista_de_sentencias'	ID si hacer salir get put_line €	fin retorna
lista_de_sentencias	ID si hacer salir get put_line €	fin retorna mientras
sentencia	ID si hacer salir get put_line	ID si hacer salir get put_line fin retorna mientras
expresiones	= [(;)
acceso_a_array	[= opl1 opl2 oprel and or entonces fin ;)] ,
parametros_llamadas	(opl1 opl2 oprel and or entonces fin ;)] ,
expresion	not - ID INTEGER REAL true false (entonces fin ;)] ,
disyuncion	not - ID INTEGER REAL true false (entonces fin ;)] ,
disyuncion'	or €	entonces fin ;)] ,
conjuncion	not - ID INTEGER REAL true false (or entonces fin ;)] ,
conjuncion'	and €	or entonces fin ;)] ,
relacional	not - ID INTEGER REAL true false (and or entonces fin ;)] ,
relacional'	oprel €	and or entonces fin ;)] ,
aritmetica	not - ID INTEGER REAL	oprel and or entonces

	true false (fin ;)] ,
aritmetica'	opl2 €	oprel and or entonces fin ;)] ,
termino	not - ID INTEGER REAL true false (opl2 oprel and or entonces fin ;)] ,
termino'	opl1 €	opl2 oprel and or entonces fin ;)] ,
negacion	not - ID INTEGER REAL true false (opl1 opl2 oprel and or entonces fin ;)] ,
factor	- ID INTEGER REAL true false (opl1 opl2 oprel and or entonces fin ;)] ,
factor'	ID INTEGER REAL true false (opl1 opl2 oprel and or entonces fin ;)] ,
array_o_llamada	([€	opl1 opl2 oprel and or entonces fin ;)] ,
opl1	* /	not - ID INTEGER REAL true false (
opl2	+ -	not - ID INTEGER REAL true false (
oprel	oprel	not - ID INTEGER REAL true false (
booleano	true false	opl1 opl2 oprel and or entonces fin ;)] ,
M	€	ID si hacer salir get put_line fin ;
lista_de_expr	not - ID INTEGER REAL true false (])
resto_lista_expr	, €])