

Estructura general del programa.-

$Prog ::= \text{program Ident PYCOMA Bloque PUNTO}$

$\text{Ident} ::= \text{id PA Iden PC}$

$\text{Iden} ::= \text{id} \mid \text{Iden COMA id}$

El programa comienza con la palabra reservada *program* seguida de un identificador de programa que puede estar parametrizado.

Estructura de los bloques.-

$\text{Bloque} ::= \text{Tvar TBloque} \mid \text{TBloque}$

$\text{Tvar} ::= \text{var Tvar2}$

$\text{Tvar2} ::= \text{id 2PUNTOS Tipo PYCOMA} \mid \text{id 2PUNTOS Tipo PYCOMA Tvar2}$

$\text{Tipo} ::= \text{integer} \mid \text{boolean}$

$\text{TBloque} ::= \text{begin TBloque2 end}$

$\text{TBloque2} ::= \lambda \mid \text{TAsig TBloque2} \mid \text{TRead TBloque2} \mid \text{TWrite TBloque2}$

Un bloque comienza con una declaración de variables (aunque esta sección puede estar vacía). A continuación, para marcar el comienzo y el fin del cuerpo del bloque se utilizan las palabras reservadas *begin* y *end*.

Estructura de las operaciones.-

TRead ::= read TA id TC PYCOMA

TWrite ::= write TA Text TC PYCOMA

Text ::= texto | id

TAsig ::= id ASIG Exp

Exp ::= ExpSimple | ExpSimple Comp ExpSimple

ExpSimple ::= ExpSimple OpAd Term | Term

Term ::= Term OpMul Fact | Fact

Fact ::= numero | true | false | id | OpUn Fact | (Exp)

OpAd ::= + | - | or

OpMul ::= * | / | and

OpUn ::= + | - | not

Comp ::= <= | >= | < | > | = | !=

La operación *read* lee una secuencia de caracteres introducida por el usuario y la almacena en una variable, mientras que la operación *write* muestra una secuencia de caracteres por pantalla.