# Syntax.in

<program> ::= *** <list_instructions> ***

<list_instructions> ::= <instruction> | <instruction><list_instructions>

<instruction> ::= <decl_instruction> |<rw_instruction>|<assign_instruction> |
<if_instruction> | <for_instruction>

<decl_instruction> ::= <var_decl_instruction> | <arr_decl_instruction>

<var_decl_instruction> ::= var space <identifier> <var_type> ;

<arr_decl_instruction> ::= var <identifier> [ <unsigned_integer> ] <var_type>;

<var_type> ::= real | integer | long_integer

<rw_instruction> ::= <rw_keywords> ( <identifier> ) ;

<rw_keywords> ::= read | write

<assign_instruction> ::= <identifier> := <expression>

<if_instruction> ::= condition ( <list_conditions> ) <program> | condition (
<list_conditions> ) <program> otherwise <if_instruction> | condition ( <list_conditions> )
<program> otherwise <program>

<list_conditions> :== <condition> | <condition> && <list_conditions>

<condition> ::= <expression> <relation> <expression>

<relation> ::= :< | :> | :== | :!=  | :>= | :<=

<operator_gr1> ::= +|-

<operator_gr2> ::= *|/

<op_const> ::= <real>|<identifier>

<expression> ::= <expression> <operator_gr1> <term> | <term>

<term> ::= <term> <operator_gr2> <factor> | <factor>

<factor> ::= ( <expression> ) | <op_const>

<for_instruction> ::= step_loop ( <identifier>, <identifier>|<constant>, <identifier>| <constant>, <constant>) <program>