///////////////////////////////////////////////////////////////////////////////

// Основа

<Program> ::= <Name main function> <Command block>;

<Command block> ::= <Start block> <Commands> <End block>;

<Commands> ::= (<Command>)+;

<Command> ::= (**<value>** | **<any branching>** | **<any loop> | <any interrupt operator>** )<Command separator>;

//

<identificator> ::= (<letter>)+ (<digit> | <letter>)\* ;

///////////////////////////////////////////////////////////////////////////////

///////////////////////////////////////////////////////////////////////////////

// Комманды

// TODO : оператор присвоения

**<value>** ::= <Call function> | **<Assign for variable>** | <variable> | <expression> | <literal>;

**<Call function>** ::= <FunctionName><list arguments>;

<FunctionName> ::= <identificator> ;

<list arguments> ::= <Start list arguments> <set arguments> <End list arguments>;

<set arguments> ::= <value> (<variable separator> <value> )? ;

**<Assign for variable>** ::= ((***<init variable>*** | <variable> )'=' (<value> | <list values>)) ;

***<init variable>*** ::= ***<name type> (*<is pointer> )?**<common separator> <identification>;

***<name type>*** ::= любой из **1 {} ;**

**1 - {**

<name integer> ::= "int" ;

<name float> ::= "float" ;

<name char> ::= "char";

<name string> ::= "string";

<name logic> ::= "bool" ;

**1 - }**

**<is pointer> ::=** <name pointer>;

<name pointer> ::= '\*';

**<variable>** ::= (***<operator dereference>)?*** ( **<array>** | <identificator>) ;// TODO : разыменование

**<operator dereference>** ::= "\*";// TODO оператор разыменования

**<array>** ::= <identification> ('[' <integer> ']')+;

**<list values>** ::= <Start block> <first value in list> (<another values in list>)\* <End block>;

<first value in list> ::= <value>;

<another values in list> ::= <variable separator> <value>;

**<literal>** ::= <number> | <char> | <string> | <logic>; // TODO : дополнить

**<any branching> ::= <usual branching> | <switch branching> ;**

**<any loop> ::= <loop with precondition>| <loop with postcondition> | <loop with counter> ;**

<reserve rule literal> ::= <Name main function>;

<Name main function> ::= "main";