

$G = ( N, \Sigma, P, S )$

$S = \text{program}$

$N = \{ \text{decllist, declaration, type, type1, listdecl, cmpdstmt, stmt, simplstmt, assignstmt, structstmt, whilestmt, ifstmt, condition, expr, term, factor, relation, operation} \}$

$\Sigma = \{ ":", ";", "(", ")", "read", "write", "int", "char", "str", "[", "]", "bool", "list", "=", "+", "*", "float", "finish", "else", "{", "}", "if", "loop", "START", "STOP", "<", "<", "<=", "<>", ">=", ">", "==", "!=" \}$

$P = \{$

$\text{program} ::= \text{"START" decllist "." cmpdstmt "END"}$

$\text{decllist} ::= \text{declaration} \mid \text{declaration decllist}$

$\text{declaration} ::= \text{IDENTIFIER "(" type ")" "."}$

$\text{type} ::= \text{type1} \mid \text{listdecl}$

$\text{type1} ::= \text{"bool"} \mid \text{"char"} \mid \text{"str"} \mid \text{"float"} \mid \text{"int"}$

$\text{listdecl} ::= \text{"list" "(" type1 ")"}$

$\text{cmpdstmt} ::= \{ \text{stmt} \}$

$\text{stmt} ::= \text{simplstmt} \mid \text{structstmt "."}$

$\text{simplstmt} ::= \text{iostmt} \mid \text{assignstmt} \mid \text{finish}$

$\text{iostmt} ::= \text{("read" identifier)} \mid \text{("write" identifier) "."}$

$\text{assignstmt} ::= \text{IDENTIFIER "=" expr "."}$

$\text{structstmt} ::= \text{cmpdstmt} \mid \text{ifstmt} \mid \text{whilestmt}$

$\text{whilestmt} ::= \text{"loop" ifstmt}$

$\text{ifstmt} ::= \text{"if" ":" condition "[" cmpdstmt "]" "."}$

$\text{condition} ::= \text{expr relation expr}$

$\text{expr} ::= \text{[expr("+" | "-")]term "."}$

$\text{term} ::= \text{term ("*" | "/" ) factor} \mid \text{factor "."}$

$\text{factor} ::= \text{(" expr") } \mid \text{IDENTIFIER} \mid \text{CONSTANT}$

$\text{relation} ::= \text{"<"} \mid \text{"<="} \mid \text{"="} \mid \text{"<>" } \mid \text{">="} \mid \text{">"}$

$\}$

Documentation:

$G$  – is the definition of the grammar  $( N, \Sigma, P, S )$

$S$  - is the starting symbol (the syntactical construct of the program)

N - is the non-terminal (declaration, statement, expression, term, factor,...)

$\Sigma$  - is the terminals (identifiers, constants, operators, separators, reserved words)

P - is the syntactical rules (BNF style)