

```

topRule = "startprogram", "variable", varsBlok, ";", "startblok", statement, "endblok";

varsBlok = "int_2", identifier, { commaAndIdentifier }; identifier = "_",
up_letter, up_letter, up_letter, up_letter, up_letter, up_letter, up_letter, up_letter;

commaAndIdentifier = ",", identifier;

statement = write | read | assignment | ifStatement | goto_statement | goto_declaration | forToOrDownToDoRule | while | repeatUntil;

read = "get", identifier;

write = "put", equation;

assignment = identifier, "<==", equation;

ifStatement = "if", (equation), statement, [ elseStatement ];

elseStatement = "else", statement; goto_statement = "goto", identifier; goto_declaration = identifier, ":";

forToOrDownToDoRule = "for", assignment, ("to" | "downto"), equation, "do", statement;

while = "while", (equation), "do", while_statement, "endwhile";

while_statement = write | read | assignment | whileifStatement | goto_statement | goto_declaration | whilefortoordowntodorule |
while | repeatUntil | continnewhile | exitwhile;

exitwhile = "exitwhile";

continnewhile = "continnewhile";

whileifstatement = "if", (equation), while_statement, [elseStatement];

whilefortoordowntodorule = "for", assignment, ("to" | "downto"), equation, "do", while_statement; repeatUntil =
"repeat", (statement), "until", equation;

equation = signedNumber | identifier | notRule, { operationAndIdentOrNumber | equation };

notRule = notOperation, signedNumber | identifier | equation;

operationAndIdentOrNumber = mult | arithmetic | logic | compare | signedNumber | identifier | equation; arithmetic = "+" | "-";

mult = "" | "/" | "%";

logic = "and" | "or";

notOperation = "!"; compare = "eq" | "noteq" | "less" | "gr";

comment = LComment, { low_letter | up_letter | digit }, ;

LComment = "/";

signedNumber = [ sign ], digit, { digit }; sign = "+" | "-";

low_letter = "a" | "b" | "c" | "d" | "e" | "f" | "g" | "h" | "i" | "j" | "k" | "l" | "m" | "n" | "o" | "p" | "q" | "r" | "s" | "t" | "u" | "v" | "w" | "x" | "y" | "z";
up_letter = "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H" | "I" | "J" | "K" | "L" | "M" | "N" | "O" | "P" | "Q" | "R" | "S" | "T" | "U" | "V" | "W" | "X" | "Y" |
"Z";

digit = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9";

```

