<Программа>::= <Функция> { <Функция> } | <Процедура> { <Процедура> } | <оператор выражения>’;’ | main ‘(’’)’ <составной оператор>

<Функция>::= <тип> <имя> <список параметров> <составной оператор>

<Процедура>::=’def’ <имя> <список параметров> <составной оператор>

<список параметров>::= ‘(‘ <переменная> { ‘,’ <переменная> } ‘)’

<Переменная>::= <тип> <имя>

<имя>::=<буква>|<имя><буква>|<имя><цифра>

<тип>::= ‘int’ | ‘boolean’ | ‘double’ | ‘string’

<константа>::= <число> | <логическое значение> | <строка>

<число>::= <знак> <цифра> {<цифра>} | <цифра> {<цифра>}’.’<цифра> {<цифра>}

<логическое значение>::= ‘true’ | ‘false’

<строка> ::= “{<символ>}“

<составной оператор>::= ‘{‘{ <оператор>{<оператор>} } ‘}’

<оператор>::= <оператор ввода> | <оператор вывода> | <оператор выражение>’;’ | <вызов процедуры> | <спец. оператор> | <вызов функции>

<оператор ввода>::= ‘read’’(‘<имя>’)’’;’

<оператор вывода>::=’write’’(‘[<имя> | <строка>]{ ‘,’ [<имя> | <строка>] }’)’’;’

<оператор выражения>::= <имя><приоритет\_1><приоритет\_2> | <переменная> ‘=’ <приоритет\_2> | <приоритет\_2>

<приоритет\_1>::= ‘=’ | ‘+=’ | ‘-=’ | ‘\*=’ | ‘/=’ | ‘%=’

<приоритет\_2> ::= <приоритет\_3> {<знак\_2><приоритет\_3>}

<знак\_2> ::= ‘||’

<приоритет\_3> ::= <приоритет\_4> {<знак\_3><приоритет\_4>}

<знак\_3> ::= ‘&&’

<приоритет\_4> ::= <приоритет\_5> {<знак\_4><приоритет\_5>}

<знак\_4> ::= ‘|’

<приоритет\_5> ::= <приоритет\_6> {<знак\_5><приоритет\_6>}

<знак5> ::= ‘^’

<приоритет\_6> ::= <приоритет7> {<знак6><приоритет\_7>}

<знак\_6> ::= ‘&’

<приоритет\_7> ::= <приоритет\_8> {<знак\_7><приоритет\_8>}

<знак\_7> ::= ‘==’ | ‘<>’

<приоритет\_8> ::= <приоритет\_9> {<знак\_8><приоритет\_9>}

<знак\_8> ::= ‘<’ | ‘>’ | ‘<=’ | ‘>=’

<приоритет\_9> ::= <приоритет\_10> {<знак\_9><приоритет\_10>}

<знак\_9> ::= ‘<<’ | ‘>>’

<приоритет\_10> ::= <приоритет11> {<знак\_10><приоритет\_11>}

<знак\_10> ::= ‘+’ | ‘-‘

<приоритет\_11> ::= <приоритет\_12> {<знак\_11><приоритет\_12>}

<знак\_11> ::= ‘\*’ | ‘/’ | ‘%’

<приоритет\_12> ::= <знак\_12><приоритет\_13> | <приоритет\_13>

<знак\_12> ::= ‘!’

<приоритет\_13> ::= <знак\_13><приоритет\_14> | <приоритет\_14>

<знак\_13> ::= ‘++’ | ‘--’

<приоритет\_14> ::= <приоритет\_15> {<знак\_14><приоритет\_15>}

<знак\_14> ::= ‘-‘ | ‘+’

<приоритет15> ::= <константа> | ‘(‘ <приоритет\_2> ‘)’ | <вызов функции> | <имя>

<вызов функции>::= <имя> ‘(‘<передаваемые параметры> { ‘,’<передаваемые параметры> }‘)’

<передаваемые параметры>::= <имя> | <константа>

<вызов процедуры>::=<имя> ‘(‘<передаваемые параметры> { <передаваемые параметры> }‘)’’;’

<спец. оператор>::= <оператор цикла> | <условный оператор> | <goto и дети>

<оператор цикла>::= <for> | <elfor> | <while> | <dowhile>

<for>::= ‘for’’(‘<переменная>’;’<оператор выражения>’;’ <оператор выражения>’)’<составной оператор> |

‘for’’(‘<имя>’;’<оператор выражения>’;’ <оператор выражения>’)’<составной оператор>

<elfor>::=‘elfor’’(‘<переменная>’;’<оператор выражения>’;’ <оператор выражения>’)’<составной оператор> <else> |

‘elfor’’(‘<имя>’;’<оператор выражения>’;’ <оператор выражения>’)’<составной оператор> <else>

<else>:= ‘else’ <составной оператор>

<while>::=’while’’(‘<оператор выражения>’)’<составной оператор>

<dowhile>::=’do’<составной оператор>’while’(‘<оператор выражения>’)’’;’

<условный оператор>::=’if’’(‘<оператор выражения>’)’ <составной оператор> <ветвление условного оператора> {<ветвление условного оператора>}

<ветвление условного оператора>::= ‘else’ <составной оператор> | ‘elseif’ ‘(‘<оператор выражения>’)’<составной оператор>

<goto и дети> ::= <return> | <break>

<return>::= ‘return’ { <оператор выражения>}’;’

<break>::= ‘break’’;’