The words - predefined tokens are specified between " and ":

Sintactical rules: (file Syntax.in)

program = "start" decllist ";" cmpdstmt ";" "end"

decllist = declaration | declaration ";" decllist

declaration = (type | array) IDENTIFIER {"," IDENTIFIER}

type = "number" | "string" | "integer" | "char" | "real"

array = "list" "[" type "]"

cmpdstmt = stmt | stmt ";" cmpdstmt

stmt = simplstmt | structstmt

simplstmt = assignstmt | iostmt

assignstmt = IDENTIFIER "=" expression

expression = expression ("+" | "-") term | term

term = term (("*" | "/") factor | "^" factor)) | factor

factor = "(" expression ")" | IDENTIFIER | CONST

iostmt = io | ";" iostmt

io = ("read" | "print") IDENTIFIER {"," IDENTIFIER}

structstmt = cmpdstmt | ifstmt | whilestmt | forstmt

ifstmt = "if" "[" condition "]" ":" stmt [("else" ":" | "else" stmt ":")] "fi"

whilestmt = "while" "[" condition "]" ":" stmt "elihw"

forstmt = "for" "[" IDENTIFIER "<-" stmt "," stmt "," stmt "]" ":" stmt "rof"

condition = simplcond | ("and" | "or") condition

simplcond = expression RELATION expression

RELATION = ["not"] "gt" | ["not"] "geqt" | ["not"] "eq" | ["not"] "leqt" | ["not"] "lt"