**Lexic:**

Alphabet:

a. Upper (A-Z) and lower case letters (a-z) of the English alphabet;

b. Underline character '\_';

c. Decimal digits (0-9);

1. Lexic:

a.Special symbols, representing:

- operators + - \* / = < <= = >=

- separators ( ) [ ] , ; space newline tab

- reserved words:

start stop array char integer boolean string appointment date let if else then read while print for from to

b.identifiers

-a sequence of letters and digits, such that the first character is a letter; the rule is:

identifier = letter | letter{letter}{digit}

letter = "A" | "B" | . ..| "Z" | "a" | "b" | ... | "z"

digit = "0" | "1" |...| "9"

c.constants

1.integer - rule:

noconst = +no|-no|no

no = digit{no}

2.character

char = 'letter'|'digit'

4.boolean

boolean = "true" | "false"

5.string

constchar = "string"

string = char{string}

char = letter|digit

6.date

day = "1" | "2" | ... | "31"

month = "1" | "2" | ... | "12"

year = no

date = day-month-year

7.apointment

apointment = string,date

**Syntax:**

program = "START" decllist cmpdstmt "STOP"

decllist = declaration | declaration decllist

declaration = "LET" IDENTIFIER type

type1 = "INTEGER" | "CHARACTER" | "BOOLEAN" | "STRING" | "DATE" | "APOINTMENT"

arraydecl = "ARRAY" "[" nr "]" "OF" type1

type = type1|arraydecl

cmpdstmt = stmtlist

stmtlist = stmt | stmt stmtlist

stmt = simplstmt | structstmt

simplstmt = assignstmt | iostmt

assignstmt = IDENTIFIER "=" expression

expression = expression "+" term | expression "-" term | term

term = term "\*" factor | term "/" factor | factor

factor = "(" expression ")" | IDENTIFIER

iostmt = "READ" | "PRINT" IDENTIFIER

structstmt = cmpdstmt | ifstmt | whilestmt

ifstmt = "IF" condition "THEN" stmt ["ELSE THEN" stmt] "STOP IF"

whilestmt = "WHILE" condition "DO" stmt "STOP WHILE"

condition = factor RELATION expression

RELATION = "<" | "<=" | "=" | "<>" | ">=" | ">"

**Token:**

start

stop

array

char

integer

boolean

string

appointment

date

let

if

else

then

read

while

print

for

from

to

+

-

\*

/

=

<

<=

=

>=

(

)

[

]

,

;

space

newline

tab

integer

character

boolean

string

date

apointment