Lab 1b) Jercan Ioana

Lexic.txt

Alphabet:

a. Upper (A-Z) and lower case letters (a-z) of the English alphabet;
b. Decimal digits (0-9);
   - Special characters: + - * / = < <= > >= == % , [] {}
c.

1.Special symbols:

- Operators: + - * / = < <= > >= == %
- Separators: space , [] {}, ()
- Reserved words: array int string char const else if then while execute read write

2. Identifiers

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

identifier = smallletter{letter | digit | “0”}

smallletter = “a” | “b” | … | “z”

letter = "A" | "B" | ... | "Z" | “a” | “b” | … | “z”

digit = “0” | "1" | “2” ...| "9"

nonZerodigit = "1" | “2” ...| "9"

3. Constants:

- Integer:
  Number_const = “0” | [“+”|”-”] nonZeroDigit {“0” | nonZeroDigit}

- Character:
  Character = “ ‘ ” {letter | digit | special_symbol} “ ‘ ”
  special_symbol = “_” | “.” |” ” | “!” | “,”
- String:

  string_const = “ \ ” “ { letter | digit | special_symbol} ” \ “ ”

Syntax.in

Program = declist cmpdstmt

decllist = declaration | declaration decllist

declaration = type IDENTIFIER ";"

vartype = "INT" | "STRING" | "CHAR"

arraydecl = "ARRAY" "[" vartype "]" "[" nr "]"

type = vartpe | arraydecl

cmpstmt = "{" 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 | term "%" factor | factor

factor = "(" expression ")" | list

list = IDENTIFIER | arrayelement | const

iostmt = "READ" "(" IDENTIFIER ")" | "WRITE" "(" list")"

structstmt = cmpdstmt | ifstmt | whilestmt

ifstmt = "IF" "(" condition ")" "THEN" stmt ["ELSE" stmt]

whilestmt = "WHILE" "(" condition ")" "EXECUTE" stmt

condition = expression RELATION expression

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

Token.in

array int string const else if then while execute read write + - * / = < <= > >= == % , [] {}

p1. Maximum of 3 numbers

```
{
int a;
int b;
int c;
{
read ( a );
read ( b );
```

```
read ( c );
int max;
max = 0;
if ( a > b ) then {
        max = a;
        if ( max > c ) {
                write ( max );
        else
                max = c;
        }
else if ( c > b ) then {
                max = c;
        else
                max = b;
        }
}
write ( max );
}
```

p2. Checking if a number is prime

```
{
int a;
int d;
{
read ( a );

d = 2;
int number;
number = a;
while ( d < number ) execute {
        if ( number % d == 0 ) then {
                prime = 0;
        else
                prime = 1;
        }

        d = d + 1;
if ( prime == 0 ) then {
        write ( “Number is not prime” );
        else
                write ( “Number is prime” );
        }
}
}
```

p3. Sum of n numbers

```
{
int sum;
int n;
```

```
int a;
{
sum = 0;
read ( n );
while ( n > 0 ) {
        read ( a );
        sum = sum + a;
        n = n-1;
}
        write ( sum );
}
p1err.
{
int !a; //identifiers cannot have as first letter a special character, only alphabet letters
string s = “aaa;

}
```