Source code:
https://github.com/915-Motoc-Simona/FLCD/tree/main/Lab3

The Symbol Table is represented using a Hash Table.

PIF is a List of Pairs of the form (token, position), where position is:

- either another Pair representing the position of the token in the Symbol Table (bucket, position in bucket) – for constants and identifiers
  * also in this case, token will be either "id" or "const", not the actual token
- either -1 – for operators/ separators/ reserved words

The scanning algorithm takes each line of the program and tries to clasify each group of characters separated by space into one of the categories: numeric constant, char constant, string constant, identifier, operator, separator, reserved word, using pre-defined regex expressions. If a token cannot be clasified, it means there is a lexical error and the scanning algorithm stops with a corresponding message, otherwise it will go to the next token/ next line.

Regex expressions:

- Only Digits Regex:
  - start with optionally a sign, then at least a digit from 1 to 9, followed by digits from 0 to 9 and then a series of separators like end of line : ; + - / % , ( ) { } [ ] ", but the last part won't be captured in the match
  - Or just 0
- Char Constant Regex:
  - start with '
  - followed by a character that is either a digit or a lowercase/ uppercase letter
  - end with '
- String Constant Regex:
  - start with '
  - followed by any number of characters that are either digits or lowercase/ uppercase letters
  - end with '
- Identifier Regex:
  - Start with a lowercase/ uppercase letter, followed by any number of characters that are either digits or letters
- Operator Regex:
  - Something that is one of the following: + - * / % < <= > >= != = ==
- Separator Regex:
  - Something that is one of the following: \n ; , ( ) { } [ ]
- Reserved Words Regex:

- One of the words: var, readInt, readReal, readString, and, AND, or, OR, not, NOT, readChar, if, else, while, write, of, array