https://github.com/cs-ubbcluj-ro/lab-work-computer-science-2024-CalinescuFlorin/tree/main/1-Mini-Language-And-Scanner/FLCD_Lab

First of all, we have a class for tokens, which has the following fields:

- Token – the string literal describing the token
- Type – the type of the token: SEPARATOR, OPERATOR, RESERVED WORD, CONSTANT, IDENTIFIER
- Symbol Table position: a pair denoting the position of the token in the symbol table

Then a class for the ProgramInternalForm, which has a list of all identified tokens.

The Scanner class has multiple methods for performing all the necessary operations:

- String readFile(String filename) – returns the contents of a file and returns a String containing all the lines
- List<String> extractTokens(String lines) – separates words by the separators and returns a list of all these tokens, including the separators
- List<ScannerToken> tokenize(List<String> tokens) – processes the provided tokens: assigns the line and column position, and concatenates string constants with double quotes into tokens, if any are present
- scan(String filename) – read content of the file, uses the methods above to obtain the tokens, then performs the token classification, assigning one of the five available types; if it cannot be classified, then it prints out the lexical error
    - for constants and identifiers, they will be inserted in the symbol table and will have a position assigned
    - for other types of tokens, they will be assigned the invalid symbol table position (-1, -1)
    - once the token is classified, it will be inserted into the program internal form
    - after all tokens have been classified, the program will write into two separate files the program internal form and the symbol table