https://github.com/the-coding-cloud/FLCD/tree/main/Lab3 - GitHub repository

Lab 3 – Documentation

Purpose

The purpose of this lab was to implement a scanning algorithm for the language specified during Lab1, using the Symbol Table from Lab 2

Implementation

The algorithm goes through the program code line by line and does the following steps:

1. Tokenize
2. Classify
3. Codify

Each line is split by spaces, separators and operators. We have to take into consideration the distinction between +/- as binary operators and unary operators (work in progress 😊)

In the classification step, I check if each token falls into one the following categories: ***separator***, ***operator***, ***keyword***, ***constant*** or ***identifier***. If the token cannot be classified, we have a lexical error and it is printed to the console which token led to the error and the line.

Then, if the token is a constant or identifier, we add it to the symbol table (if it is not already there) and then add the corresponding code of the token + its position in the symbol table to the PIF.

When a Scanner object is instantiated, the separators, operators and keywords of the language are loaded into the “tokenCodes” field. At the end, the PIF and ST will be written to the files called PIF.out, respectively ST.out