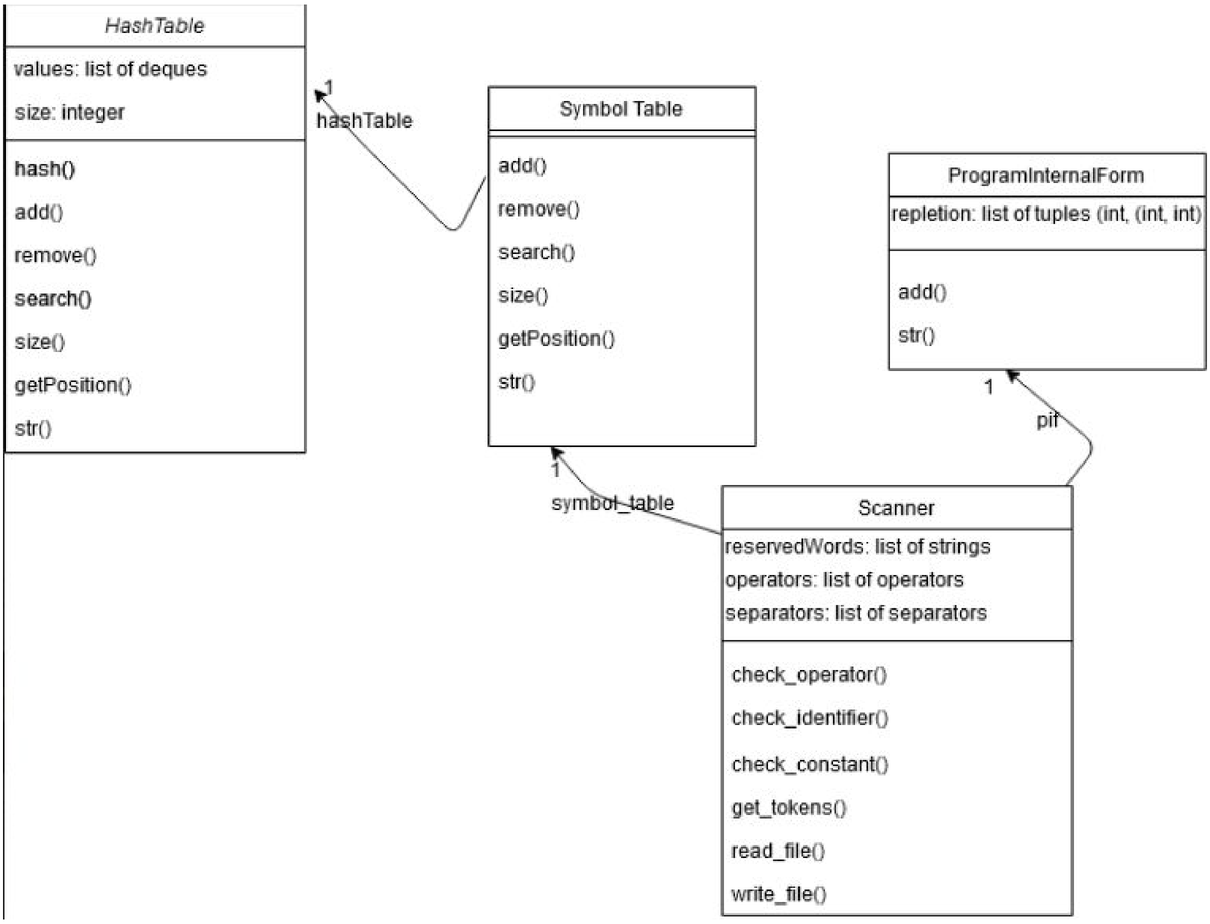
https://github.com/GeorgianaLoba/FLCD

Statement: Implement a scanner (lexical analyzer): Implement the scanning algorithm.

Input: Programs p1/p2/p3/p1err and token.in (see Lab 1a)

Output: PIF.out, ST.out, message “lexically correct” or “lexical error + location”



In order to add correct data after the lexical analysis of the program into my symbol table and program internal form, I parsed each line and checked character by character. For identifiers and constants, I used regex matching.

As I parse each line - in the read_file() function - , I call a tokenizer function - get_tokens() - that will split the line according to my given separators, identifiers, constants and so on and so forth. This will return a list of tokens that is then further processed in the read_file function and adds corresponding data to the symbol table (identifiers and constants) and program internal forms (ids, tokens, separators, reserved words).