Documentation

Github link: https://github.com/AmaliaDuma/Formal-Languages-and-Compiler-Design/tree/main/Labs/Lab_2%20-%20Scanner

Symbol Table

My symbol table is based on a hash table and has the following methods:

- __init__(self, size) : called when a new instance is created
- hash(self, element) : returns the hash value for a given element; the hash function calculates the sum of ascii codes of the characters % size of the table
- size(self) : returns the size of the table
- search(self, element) : searches for an element and returns its position or -1 if not found
- add(self, element) : adds a new element and returns its position if not already there, otherwise returns that element position
- __str__(self) : used for printing the table

Program Internal Form

A structure that holds pairs of the form (token, position in ST) and has the following methods:

- __init__(self) : called when a new instance is created
- add(self, token, pos) : adds a new pair of the form (token, position in ST) in the list
- __str__(self) : used for printing the content of the list

Scanner

A class that has holds a symbol table, a structure for the program internal form that implements the scanning algorithm and has the following methods:

- _isIdentifier(self, token) : checks if the given token is an identifier and returns true | false
- _isConstant(self, token) : checks if the given token is a constant and returns true | false

- _isOperatorPart(self, char) : checks if the given char is a part of an operator and returns true | false
- _getOperator(self, line, index) : find the next operator in the given line using the index (crt position in line) and going character by character
- _getStringConst(self, line, index) : finds the next string constant in the line using the index (crt position in line) and going character by character
- tokenize(self, line) : splits the given line into tokens and adds them to a list that is returned
- scanFile(self, filename) : scans the file and applies the scanning algorithm; writes in “st.out” the content of the symbol table and in “pif.out” the content of the structure that wraps the program internal form