

Project documentation

<https://github.com/VasilutaAndrei/LFTC/tree/master/Lab3>

1. Project requirements

Based on the specification of the mini-language chosen in the first part of the lab 1 implement a scanner that will take as input a text file containing the source program and will produce as output the following:

- PIF -> Program Internal Form
- ST -> Symbol Table

In addition, the program should be able to determine the lexical errors, specifying the location, and if possible, the type of the error.

Some additional restrictions:

- a) arbitrary length, no more than 250 characters
- b) unique for identifiers and constants
- c) hashing table

2. Used technologies & Data structures

I chose to use Python as my programming language of choice for this lab and PyCharm as IDE.

Regarding the data structures used in the application I had to use a lexicographically sorted table for my ST (Symbol Table) representation, and each element is a tuple of (variable, unique_code).

I declared my PIF (Program Internal Form) as a list of tuples, each element of the list is a tuple of (code, id) form. The code is for separators/operators/reserved words is the value from its corresponding key in the codification table and for identifiers and constants it is the second value from its corresponding tuple in the symbol table representing a unique code. The id is either -1 for separators/operators/reserved words, 0 for identifier or 1 for constant.

The special symbols as separators, operators and reserved words are represented in a list.

The codification table is a dictionary in which the key is either a separator/operator/reserved word or an identifier/constant and the value is a natural number for the first category greater than 1. For identifier it is 0 and for constant it is 1.

3. Project description

I started reading my language instructions from the file line by line then I continue by splitting the line in tokens using the function tokenize which will give constantly one token until the line ends.

If the given token is an operator, a separator or a reserved word I added code -1 to the PIF. Next, we check if our token is an identifier or a constant and in both cases we add in the ST and with the id code generated we add in the PIF with its corresponding identifier/constant code which is either 0 or 1 and with the id from the ST.