Lab 4

Github Link

https://github.com/rusuraluca/lftc/tree/main/lftc_lab4

Docs

`enum TokenTypes`: for defining token types

`class LexicalAnalyzer`: for implementing the scanning algorithm

- `constructor`
  - initializes the analyzer with the source code file and token file paths.
- `scan_file`
  - analyzes the source code, generates PIF, and manages symbol tables
- `get_tokens`
  - parses the token definitions from the token file and populates dictionaries for reserved words, operators, and separators
- `is_identifier`
  - checks if a string is a valid identifier
- `is_constant`
  - checks if a string is a valid constant

`usage`

1. instantiate a `LexicalAnalyzer` object with the source code file path and the token file path
2. call the `scan_file` method to perform lexical analysis, and generate PIF

3. results are saved in 'ST.out' (symbol tables) and 'PIF.out' (Program Internal Form) output files but also outputted in the console

```
scan = LexicalAnalyzer('p1.txt', 'token.in')
scan.scan_file()
```

- __init__:
  - initializes the analyzer with the source code file and token file paths.
- scan_file
  - analyzes the source code, generates PIF, and manages symbol tables
- get_tokens
  - parses the token definitions from the token file and populates dictionaries for reserved words, operators, and separators
- is_identifier(string)
  - checks if a string is a valid identifier
- is_constant(string)
  - checks if a string is a valid constant