Github link:

https://github.com/x64alex/Compilers/tree/main/Lab8

Structure SymbolEntry
Represents an entry in the symbol table.

Structure PIFEntry
Represents an entry in the Program Internal Form (PIF).

Function findDuplicateIndex
Find the index of a symbol in the symbol table.

Function addToSymbolTable
Adds a new entry to the symbol table or retrieves the existing index if the symbol is already present.

Function addToPIF
Adds a new entry to the Program Internal Form (PIF).

The first lex section declares regular expression patterns for various token types:

- **letter**: Matches any uppercase or lowercase alphabetical character.
- **nonnulldigit**: Matches any digit from 1 to 9.
- **digit**: Matches any digit from 0 to 9.
- **number**: Matches a sequence of digits, allowing for both non-zero digits and zero.
- **string**: Matches a string enclosed in double quotes, allowing for any combination of letters and digits within.
- **char**: Matches either a letter or a digit.
- **identifier**: Matches a valid identifier, starting with a letter followed by any combination of letters and digits.
- **identifierwrong**: Matches an invalid identifier pattern, starting with one or more digits followed by a mix of letters and digits.

The second section consists of rules for specific tokens:

- **Reserved words** like "cat", "main", etc., are matched and added to the PIF with an index of -1.
- **Arithmetic and logical operators** such as "+", "-", "*" are matched and added to the PIF with an index of -1.
- **Separators** like "[", "]", ";", etc. are matched and added to the PIF with an index of -1.
- **{identifier}** rule recognizes valid identifiers and adds them to the symbol table and PIF.
- **{identifierwrong}** rule catches invalid identifiers, prints an error message, and exits the program.
- **{number}** and **{string}** rules handle numeric and string literals, respectively, adding them to the symbol table and PIF.
- **Patterns for whitespace and newline** characters are defined to be ignored.

- **The final rule** (. { exit(1); }) captures any other unrecognized characters, leading to an error and program exit.

Function printSymbolTable
Prints the reversed symbol table to the console.

Main Program (main)
The main program initializes the lexer (yylex) to read tokens from the input file specified in the command line arguments or from standard input. It continues lexing until the end of the file is reached. After lexing, the program prints the reversed symbol table and reversed PIF using stack-based reversal techniques. Finally, memory cleanup is performed for both the symbol table and the PIF entries.