Course Project

First Deliverable: The Lexer

Quin'darius Lyles-Woods
qlyleswo@students.kennesaw.edu

Concepts of Programming Languages
Professor Jose Garrido
Section W01
4308

Bachelors of Computer Science
Kennesaw State University
1100 South Marietta Pkwy SE
Marietta, GA 30060
September 27, 2021

Task

The development of an interpreter for a subset of **Basic Language**.

Project Goals

- Process a **Basic Language** source code file.
- Tokenize the source code file.
- Detect syntactical error.
- Display appropriate error messages during runtime.

Deliverable Goals

- Develop a grammar for the subset of **Basic Language** that I will be using.
- Arrays for the *Keywords* and *Identifiers*
- Ability to input source code to lexer.
- Output tokens scanned.

Subset of the Basic Language

The subset of basic that I want to define for the course project is going to be kept as minimal as possible to focus on the process of developing an interpreter. With that in mind the lowest we can go is turing complete of course. For this to be true the language doesn't need much.

- Recursive Operators
  - `FOR...TO...NEXT`
- Conditional Jumps
  - `IF...THEN...{ELSE}`
  - `GOTO`
- Variables
  - `LET`

Tokens

- LET
- IF
- FOR
- GOTO
- +
- -
- *
- /
- =

Backus Normal Form of Basic Subset

$$
\begin{aligned}
BASIC\ PROGRAM &::= \\
&\mid EXPRESSION, EXPRESSION \\
&\mid EXPRESSION \\
EXPRESSION &::= \\
&\mid LET \\
&\mid IF \\
&\mid FOR \\
&\mid GOTO \\
&\mid IDENTIFIER\ +\ EXPRESSION \\
&\mid IDENTIFIER\ -\ EXPRESSION \\
&\mid IDENTIFIER\ *\ EXPRESSION \\
&\mid IDENTIFIER\ /\ EXPRESSION \\
LET &::= \\
&|IDENTIFIER = EXPRESSION \\
IDENTIFIER &::= \\
&\mid [a - zA - Z] \\
&\mid [a - zA - Z], [a - zA - Z] \\
IF &::= \\
&\mid EXPRESSION\ then\ STATEMENT \\
&\mid EXPRESSION\ then\ STATEMENT\ else\ IF \\
FOR &::= \\
&\mid EXPRESSION\ to\ EXPRESSION\ STATEMENT\ next \\
GOTO &::= \\
&\mid [0 - 9] \\
&\mid [0 - 9], [0 - 9]
\end{aligned}
$$