Surafel Assefa

CS 4308/02

Concepts of Programming Languages

Julia Interpreter Project – Deliverable 1 Scanner

10/18/2021

Sharon Perry

Status 100% Complete and Working as Designed

This deliverable is contained in the *JuliaScanner* class. The *JuliaScanner* class itself is stateless, rather is it an encapsulation of the lexing functionality implemented in *lexSource*. The implementation being contained in *lexSource* allows for the lexing to be done concurrently.

The lexical analysis is implemented by analyzing the source code per-char from the *PushbackReader*. Lexemes are built in two stages. First, the source file is effectively tokenized by delimiting out whitespace characters. The lexeme is then validated to match syntax rules of identifiers, digits, and keywords. An attempt is made to find the token for that lexeme. If no token is found for that lexeme, then the lexeme is derived into derivative characters and tokens are found for each character.

For each lexeme encountered that successfully receives a token, the lexeme and token are then added into a new *LexicalUnit* object, and that new *LexicalUnit* object is added to a list. After that, the list is parsed for any identifier tokens. Identifiers found in the lexical unit list is the added to a list of *IdentifierDescriptor* objects. *List< IdentifierDescriptor>* serves as a symbols table for identifiers in the source code. Both lists are contained in the *JuliaScannerResults* object which is returned by the *lexSource* method.

Graphical user interface, text

Description automatically generated

Graphical user interface

Description automatically generated with medium confidence

Graphical user interface

Description automatically generated