fns: Syntax Analysis Phase

Using **BISON** .

SYNTAX ANALYSIS

- This is the second phase in compiler construction.
- It takes the token stream generated by the lexer as input and gives the parse tree as output.
- The parser can either be handwritten or generated using tools like yacc, bison or antlr(depending on what has been used while generating the lexer).
- The syntax tree made from this is the passed onto for the semantic analysis.

BISON

- Bison is a general-purpose parser generator that converts an annotated context-free grammar into a deterministic LR or generalized LR (GLR) parser employing LALR(1) parser tables.
- It takes the .y file as an input.
- Gives .tab.h and .tab.c files as an output
- Lexer uses the .tab.h file generated here to understand the token generated is valid.

Commands Used

- bison -d grammar.y // this generates grammar.tab.c and grammar.tab.h
- flex lexer.l // this will generate lex.yy.c
- gcc lex.yy.c grammar.tab.c -ll // this generates a.out
- ./a.out < file.fns // this generates the output as token stream and gives syntax error if present.

Design Overview

- All the valid tokens are initialized in the starting.
- We have all the grammar rules, including all the non-terminals and terminals.
- Lastly there is the error printing.

The tokens stream is matched with rules and parsed. If any error is found the error message is printed and parsing is stopped.

Member Contribution