ID: 71099234

Following is the list of features that have been implemented so far in the current version of the compiler.

- **Tokenizer**: I have implemented a tokenizer that returns a token when the *next_token()* method is invoked.

- **Parser:** The parser can parse all the grammar rules. In addition to that, the parser is also able to detect scope and handle all declarations within a scope. It can detect the following errors in the input language:
  - Undefine functions or variables.
  - Multiple declarations of a function or variable in the same scope
  - Unexpected token where it is expecting a valid token.
  - Any unrecognized symbol that is not part of the grammar.
  - The output messages indicate the line number where the error occurred.

- **Code Generator:** The current version of the compiler generates code for the if statement and while statement. It is also able to generate code for nested if statements or while statements. However, it fails to update the phi function properly in the outer block. It eliminates similar instruction within a basic block but not across the program.

- **Graph Visualization:** I have implemented CFG visualization in my compiler. The compiler produces dot language in a graph.dot file. This file can be used to visualize the graph in any graph visualization tool.