COMPILER CONSTRUCTION

LAB TERMINAL

NAME:

Muhammad Aamir Bin Habib (FA20-BCS-010)

CLASS:

BCS-7B

Submitted To:

Mr. Bilal Haider

Date:

27-December -2023

//

QUESTION NO.4

Step-by-Step Process of the Mini Compiler:

1. Input:

Input: Java source code.

2. Scanner (Lexical Analysis):

Tokenize the code and remove comments and whitespace.

Steps:

- Read the input Java source code.
- Break down the code into tokens (keywords, identifiers, literals, operators, etc.).
- Eliminate comments and whitespace to obtain a stream of relevant tokens.
- Identify and categorize each token according to the rules of the Java language.

3. Semantic Analysis:

Check the code for adherence to language rules, proper variable usage, and type compatibility.

Steps:

- Receive the stream of tokens from the Scanner.
- Analyze the structure of the code to ensure it follows the syntactic rules of Java.
- Perform semantic checks, including:
- Verify proper variable declarations and usage.
- Check for type compatibility.
- Ensure that identifiers are declared before use.
- Report errors for any violations of semantic rules.

4. Memory Analyzer:

Focus on memory-related aspects, checking for allocation and deallocation issues.

Steps:

- Receive the analyzed code from the Semantic Analysis phase.
- Perform memory-related checks, including:
- Track variable usage and ensure proper initialization.
- Check for memory leaks by verifying proper deallocation.
- Manage dynamic memory allocation (if applicable).
- Report errors or warnings related to memory issues.

5. Output:

Report any errors or warnings found during the scanning, semantic analysis, and memory analysis phases.

Steps:

- Generate a comprehensive report based on the findings of the Scanner, Semantic Analysis, and Memory Analyzer.
- Output any errors, warnings, or messages indicating the status of the input Java code.
- Provide a summary of the code's correctness, adherence to language rules, and memory-related issues.