Logo, company name

Description automatically generated

**COMSATS University Islamabad (CUI)**

**Lab Terminal**

**Submitted to: Sir Bilal Bukhari**

**Submitted By: Muneeb Ahmed Ayub**

**Reg No: FA21-BCS-051**

**Course: Topics in Computer Science**

**Date: 3 jan, 2025**

**Questions 2:**

1.⁠ ⁠Tokenize Function (Lexer)

The tokenize function in the lexer is responsible for breaking down the input source code into meaningful symbols or tokens. This process is crucial as it translates the source code into a format that can be further analyzed by the parser.

Function Description:

Input: A string containing the source code.

Process: The function scans the input string character by character, identifying patterns that match predefined token types (e.g., keywords, identifiers, literals, operators).

Output: A list of tokens, where each token is a tuple containing the type of the token and its value.

Example Function Signature:

java

public List<Token> tokenize(String sourceCode) {

// Implementation of the function to scan and recognize tokens

}

2.⁠ ⁠Parse Function (Parser)

The parse function in the parser is responsible for converting the list of tokens produced by the lexer into an Abstract Syntax Tree (AST). This tree represents the hierarchical structure of the source code, making it easier to analyze and generate code.

Function Description:

Input: A list of tokens produced by the lexer.

Process: The function uses a set of grammar rules to recursively analyze the token list, building a tree structure that represents the syntactic structure of the source code.

Output: An Abstract Syntax Tree (AST), where each node represents a construct occurring in the source code.