**Explain 2 code functions of your mini compiler**

**Lexical Analyzer (**lexer**)**:

* **Functionality**: Breaks the input code into tokens such as keywords, identifiers, operators, and literals.
* **Example**

int lexer(const char \*sourceCode) {

    // Tokenize code and return tokens for the parser.

}

**Parser (**parse**)**:

* **Functionality**: Parses tokens and constructs a syntax tree to ensure the code follows grammar rules.
* **Example**:

TreeNode \*parse(TokenList tokens) {

    // Build and return an abstract syntax tree.

}