Logo, company name

Description automatically generated

**COMSATS University Islamabad (CUI)**

**Lab Terminal**

**Submitted to: Sir Bilal Bukhari**

**Submitted By: Sohaib Safeer**

**Reg No: FA21-BCS-054**

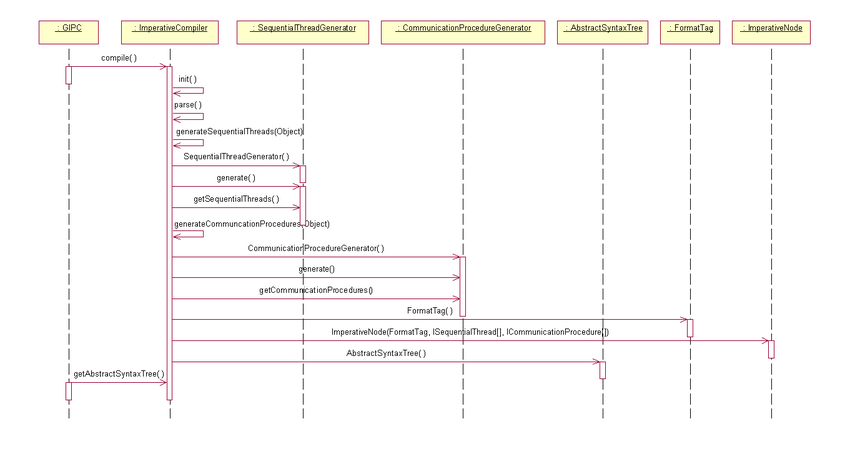
**Course: Topics in Computer Science**

**Date: 3 jan, 2025**

**Question 1: Briefly explain your mini compiler**

This mini compiler is designed to translate a subset of the C programming language into an intermediate representation, which can then be executed by a virtual machine or translated further into machine code.

**q2: Draw koi bhi diagram of your compiler (Use case etc)**



This diagram illustrates a basic compiler pipeline:

1. **Source Code:** The input to the compiler.
2. **Lexical Analyzer:** Breaks down the source code into a stream of tokens (e.g., keywords, identifiers, operators, literals).
3. **Syntax Analyzer (Parser):** Analyzes the token stream to ensure it conforms to the language's grammar. Constructs an Abstract Syntax Tree (AST) or other intermediate representation.
4. **Semantic Analyzer:** Performs type checking, scope resolution, and other semantic checks to ensure the program is meaningful and correct.
5. **Intermediate Code Generator:** Translates the AST into an intermediate representation, such as three-address code or assembly-like instructions.
6. **Optimizer (Optional):** Applies optimization techniques to improve the performance of the generated code (e.g., constant folding, dead code elimination).
7. **Code Generator:** Translates the intermediate code into machine code for a specific target architecture