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**

**q4: with explain of optimization in your mini compiler**

While the provided mini-compiler doesn't explicitly include optimization techniques, here are some common optimizations that could be incorporated:

* **Constant Folding:**
  + **Concept:** Evaluate constant expressions at compile time instead of runtime.
  + **Example:**
    - x = 2 + 3;
    - The compiler would directly replace this with x = 5;
  + **Benefits:** Reduces runtime calculations and improves performance.
* **Dead Code Elimination:**
  + **Concept:** Remove code that has no effect on the program's output.
  + **Example:**
    - If a variable is assigned a value but never used, the assignment and any related code can be removed.
  + **Benefits:** Reduces code size and improves execution speed.
* **Common Subexpression Elimination:**
  + **Concept:** Identify and remove redundant computations.
  + **Example:**
    - If the same expression a + b appears multiple times, calculate it once and store the result for reuse.
  + **Benefits:** Reduces the number of calculations and improves performance.
* **Strength Reduction:**
  + **Concept:** Replace expensive operations with cheaper ones.
  + **Example:**
    - Replace multiplication by a power of 2 with a left shift operation.
  + **Benefits:** Can significantly improve performance, especially for computationally intensive tasks.

**Implementation Notes:**

* These optimizations would typically be implemented in the **optimizer** stage of the compiler, after the intermediate code generation.
* The effectiveness of these optimizations depends on the complexity of the source code and the target architecture.

By incorporating these optimization techniques, the mini-compiler could generate more efficient and optimized code, leading to improved performance of the compiled programs.