

**Lab Terminal**

**Name : Muhammad Wasiq**

**Reg. no . : fa21-bcs-039**

**Course : Compiler Construction**

**Q.4:**  
**Input and Output for the Mini C++ Compiler**

#### 

#### 1.Input Code (C++ Source Code):

int main() {

return 42;

}

### Step-by-Step Output

#### **1. Lexical Analysis Output (Tokens)**

The compiler breaks the input code into tokens:

Tokens:

(KEYWORD, int)

(IDENTIFIER, main)

(SYMBOL, ()

(SYMBOL, ))

(SYMBOL, {)

(KEYWORD, return)

(NUMBER, 42)

(SYMBOL, ;)

(SYMBOL, })

#### **2. Syntax Analysis Output (AST)**

The compiler constructs an Abstract Syntax Tree (AST) from the tokens:

AST:

{type: return, value: 42}

#### **3. Semantic Analysis Output**

The compiler validates the AST for semantic correctness:

Semantic Analysis Passed!

### **Overall Output:**

Tokens:

(KEYWORD, int)

(IDENTIFIER, main)

(SYMBOL, ()

(SYMBOL, ))

(SYMBOL, {)

(KEYWORD, return)

(NUMBER, 42)

(SYMBOL, ;)

(SYMBOL, })

AST:

{type: return, value: 42}

Semantic Analysis Passed!

If there are errors in the input code (e.g., missing semicolon or invalid return type), the compiler will output detailed error messages during the respective phase (lexical, syntax, or semantic analysis). Would you like to see examples of handling errors or invalid input?