**Group 7 Team Member Names:**

Ashley Finn (finnac)

Jacob Costello (costeljt)

Triet Pham (phamt7)

Ethan Link (linkeh)

**Github Link:** https://github.com/ethanlink255/Lexer\_Parser

**YouTube Link:**

**Project Responsibilities**

Ethan Link: Coding work on the new features.

Triet Pham: Coding work on the new features.

Jacob Costello: Documentation and deliverable document.

Ashley Finn: Documentation and deliverable document

**Features added:**

**Binary Operator:** XOR, a binary operator that returns true if and only if its arguments differ.

**Type:** Double: added the double type to CLite. Similar to float in that it deals with decimal numbers.

**Control structure:** Added the for loop control structure that executes the statement repeatedly depending on the input.

**CLite Documentation Appendix Additions:**

**A.1 Lexical and concrete syntax of CLite:**

**Type *→* int | bool | float | char | double**

***Double → Integer . Integer***

***Literal → Integer*** | ***Boolean* | *Float* | *Char* | *Double4***

***ForStatement →*** *for (****Assignment***; ***Expression; Expression***)*Statement*

**A.2 Lexical and concrete syntax of CLite:**

***BooleanOp = && | || | ^^***

***Value = intValue | BoolValue | FloatValue | CharValue | DoubleValue***

***DoubleValue = Double*** *doubleValue*

**A.3 Type System of CLite:**

**Type Rule A.4:**

**2 (c) *If the type of its*** target ***Variable is*** float***or*** double*,* ***then the type of its source Expression must be one of the following types:*** float, int, ***or*** double.

**Type Rule A.5:**

**3 (b) *If its BinaryOp*** op***is arithmetic (*** +, - , \* , /  ***) Then both its Expressions must be one of the following types:*** int, float, ***or*** double.

***3 (d) If*** *op* ***is boolean ( &&, ||, ^^) then both its Expressions must be*** *bool.*

**Type Rule A.6:**

***3 (b) If the Operator is relational (****<, <=, >, >= , ==, !=****) or boolean (****&&, ||, ^^****), then its result type is*** *bool.*

**A.4 Semantics of CLite:**

**Meaning Rule A.8:**

**3. *If the operator is a floating point operator, then floating point arithmetic using the IEEE standard is performed on the*** float ***operands, resulting in a*** float ***result. If the operator is a double-precision floating-point operator, then double-precision floating point arithmetic using the IEEE standard is performed on the*** double ***operands resulting in a*** double ***result.***

**5 (c) The operator ^^ is interpreted as:**

***a*  ^^ *b* ≡ if *a* then *b* else *true***

**Meaning Rule A.9:**

***3 If the operator is*** *int-/float-/double-****, then the integer/floating point/ double operand’s sign is inverted.***