1. Summary
   1. Program that can take an input file written in Lindy and execute appropriate instructions. Executes as expected according to simple defined Lindy syntax handling general purpose programming concepts including scoped variables, literals, function calls, loops, and conditional statements.
2. Must Have Requirements
   1. Variables
      1. Handles variable initialization
         1. A variable will be initialized with a value and then printed
         2. Variable value prints to the screen
      2. Handles valid variables references
         1. An existing variable will be printed
         2. Variable value prints to the screen
      3. Handles invalid variable references
         1. A non-existent variable will be printed
         2. An error message is given
      4. Handles variable reassignment
         1. An existing variable will be given a new value of the same type and printed
         2. New value prints to the screen
      5. Handles variable type reassignment
         1. An existing variable will be given a new value of a different type and printed
         2. New value prints to the screen
      6. Handles user output
         1. Print function will be called
         2. Arguments of print will be displayed to the user
      7. Handles input
         1. Prompt function will be called and value passed to print
         2. user will be able to enter input and that input will show to the screen after
      8. Supports integers
         1. Variable will be assigned to a whole number and printed
         2. Number will be displayed
      9. Supports floating point numbers
         1. Variable will be assigned to a floating point number and printed
         2. Number will be displayed
      10. Supports Booleans
          1. Variable will be assigned to True and printed
          2. True will be displayed
          3. Variable will be assigned to False and printed
          4. False will be displayed
      11. Supports strings
          1. Variable will be assigned to a text string and printed
          2. String text value will be displayed
      12. Supports lists
          1. Variable will be assigned to a list of numbers and printed
          2. List of values will be displayed
             1. Repeat for different types of variables
             2. Repeat with mixed variable types
      13. Handles invalid types
          1. Variable will be assigned to an unknown value type
          2. Error message will be displayed
   2. Conditional branching
      1. Handles true if statements
         1. An if statement with a true condition containing a print statement will be executed
         2. The message from print will be displayed
      2. Handles false if statements
         1. An if statement with a false condition containing a print statement will be executed
         2. The message from print will not be displayed
      3. Handles true else statements
         1. An else statement with a true condition, attached to an if statement with a false condition, containing a print statement will be executed
         2. The message from print will be displayed
         3. An else statement with a true condition, attached to an if statement with a true condition, containing a print statement will be executed
         4. The message from print will not be displayed
      4. Handles false else statements
         1. An else statement with a false condition, attached to an if statement with a false condition, containing a print statement will be executed
         2. The message from print will not be displayed
      5. Enforces if syntax
         1. An if statement will bad syntax will be executed
         2. An error message will be displayed
      6. Enforces else syntax
         1. An else statement not attached to an if statement will be executed
         2. An error message will be displayed
      7. Handles multi-level if statements
         1. A true if statement containing a print statement inside a true if statement will be executed
         2. The message from print will be displayed
         3. A false if statement containing a print statement inside a true if statement will be executed
         4. The message from print will not be displayed
         5. A true if statement containing a print statement inside a false if statement will be executed
         6. The message from print will not be displayed
   3. Loops
      1. Handles true loop execution
         1. A loop with a true condition will contain print statements
         2. Print messages will be displayed until condition is false
      2. Handles false loop execution
         1. A loop with a false condition will contain print statements
         2. Print messages will not be displayed
      3. Enforces loop syntax
         1. Loop with bad syntax will be executed
         2. An error message will be displayed
      4. Handles multi-level looping
         1. A loop inside of a loop will be executed to display messages
         2. Messages will be displayed predictably until loop conditions are false
   4. Functions
      1. Supports valid function calls
         1. A function containing a print statement will be called between print statements
         2. The messages should be displayed in the expected order
      2. Handles invalid function calls
         1. A non-existent function will be called
         2. An error message will be displayed
      3. Supports returns with value
         1. A function containing a value return before function print statements will be called
         2. The value should be returned and trailing function code should not execute
      4. Supports returns without value
         1. A function containing a valueless return before function print statements will be called
         2. No value should be returned and trailing function code should not execute
      5. Supports returns from if statements
         1. A return will be made from within an if statement
         2. Trailing if and function code will not execute
      6. Supports returns from loops
         1. A return will be made from within a loop
         2. Trailing loop and function code will not execute
      7. Enforces return syntax
         1. A return will bad syntax will be made
         2. An error message will be displayed
   5. Scope
      1. Handles in scope variables
         1. In scope variable will be referenced by print
         2. Variable value will be displayed
      2. Handles out of scope variables
         1. Out of scope variable will be referenced by print
         2. An error message will be displayed
   6. Source files
      1. Executes source files
         1. A source file will be given only containing valid code will be executed
         2. Code will execute as expected and no error messages will be displayed
      2. Handles bad input
         1. A source file with invalid code will be executed
         2. Code will execute as expected until the invalid code, at which point an error message will be given and execution will halt
   7. Arithmetic
      1. Handles addition
         1. Print will be called on addition of two numbers
         2. Proper answer will be displayed
      2. Handles subtraction
         1. Print will be called on subtraction of two numbers
         2. Proper answer will be displayed
      3. Handles multiplication
         1. Print will be called on multiplication of two numbers
         2. Proper answer will be displayed
      4. Handles division
         1. Print will be called on division of two numbers
         2. Proper answer will be displayed
      5. Handles modulo
         1. Print will be called on a number modulo another
         2. Proper answer will be displayed
      6. Handles exponents
         1. Print will be called on a number raised to the power of another
         2. Proper answer will be displayed
      7. Handles order of operations
         1. Print will be called on an equation containing mixed operands
         2. Proper answer will be displayed
      8. Handles parenthesis
         1. Print will be called on an equation containing mixed operands and parenthesis to denote a different order of operations
         2. Proper answer will be displayed
3. Desired Extras
   1. Documentation
      1. Covers syntax topics
   2. Functions
      1. Handles recursion
         1. A recursive function will be called
         2. The function will execute as expected
      2. Handles inner functions
         1. A function containing an inner function will be called
         2. The function should execute normally
         3. The inner function will be called from non-parent function
         4. An error message will be displayed
   3. Interpreter
      1. Handles proper code
         1. An interpreter will start without file input and accept valid syntax from user
      2. Enforces syntax
         1. Bad syntax will be input by the user
         2. Error message will be displayed without exiting the program and ready for additional user input
      3. Controlled exit
         1. The interpreter will only exit when given proper command from the user
   4. Variables
      1. Supports inner lists
         1. Set a variable to a list containing lists of numbers
         2. Iterate through each number using nested loops and print each value
         3. Each value should be displayed properly