**TASK**: Look at each of the phrases below and ensure you understand what each of these means. For any that you do not understand, do a little research to find a definition of each term. This research may involve looking back over these notes, or the associated lecture notes. It may also involve searching for these terms on the Internet.

Source code: They are the set of instructions which can be read by humans and are used in

Programming languages like python, java .example a python script that performs

Calculations.

Machine code: They are low level binary code and consists of (0s and 1s) specific to

Architecture of computer’s CPU. Example: Instructions like 10110000 01100001

Interpreter: They are the program that executes source code and translates it into machine code

Compiler: They are the program that converts source code into machine code before execution.

Example: C++ uses a compiler (like g++).

2GL (Second Generation Language): Assembly languages that provide a way to write machine

instructions in a more human-readable format.

3GL (Third Generation Language): High-level programming languages that are more abstract

and user-friendly, like C, Java, and Python.

4GL (Fourth Generation Language): Even higher-level languages designed to be closer to

human language and often include database management

capabilities (e.g., SQL).

Executable: It is a compiled file that can be run directly by the computer's operating system.

Example: A .exe file.

Expressions: They are mixtures of variables, operators, and values that gives a result when

evaluated.

Example: 2 + 3 \* 4 evaluates to 14.

Operators and Operands:

Operators: Symbols that specify operations to be performed (e.g., +, -, \*, /).

Operands: The values or variables that operators act upon.

Example: In the expression 5 + 3, + is the operator, and 5 and 3 are the

operands.

Syntax Errors: Syntax errors occur when the code does not match the rules of the

programming language, preventing it from being compiled or executed.

Example: Missing a semicolon at the end of a line in C++.

Logical Errors: Logical errors are mistakes in the code that produce incorrect results.

Example: Using the wrong formula to calculate a value.