CODE DOCUMENTATION

Question 1:

Code structure:

We structured our code in the following way:

- The code is composed of two classes: “DriverANDGate” and “AND_Gate”
- “DriverANDGate” class which contains the main method, which acts as the program’s entry point
    - Inside the main method, the output of the AND Gate neural network is printed for all possible binary input combinations.
    - The “process()” method in the AND_Gate class is invoked to obtain the output of the input combination
- “AND_Gate” class which has the same functionality of the AND gate neural network
    - This class contains several private static variables that represent the biases and weights used in the network
    - The “HiddenCalculation()” method performs the calculation at each node of the network, and returns the calculated value after applying an activation function.
    - The “process()” function that processes the input values and calls the “HiddenCalculation()” method to calculate the output at each node of the neural network

Code functionality:

The functionality of our code is as follows:

- The code implements a simple AND gate neural network and by performing a logical AND operation on two binary inputs (0 or 1) and returns the result.
- The “HiddenCalculation()” method that performs the calculation at every single node
    - Takes in four integer parameters: “weight”, “bias”, “input1”, and “input2”
    - It performs the calculation required at each node of the network using the parameters provided
    - The calculation is done by multiplying the weights with the inputs then adding the bias, and obtaining the result “x”
    - The function also contains a segment that acts as an activation function and decides whether the node should activate or not
        - If “x” is equal to 0, the method will return 1

        - Otherwise, it returns 0
- The “process” method is responsible for processing the input binary values through the AND gate neural network
    - It takes in two parameters “input1” and “input2” which represent the binary inputs of the AND gate
    - The method calls the “HiddenCalculation()” twice, passing the appropriate weights, biases, and input values for the hidden layer nodes(“h1” and “h2”)
    - The method calls the “HiddenCalculation()” one last time, by passing the weight, bias, and the outputs from the hidden layer as inputs to calculate the final output of the AND gate
    - The method returns the final output value