

Assignment 1: Pseudocode Development

Task: Write a detailed pseudocode for a simple program that takes a number as input, calculates the square if it's even or the cube if it's odd, and then outputs the result. Incorporate conditional and looping constructs.

Answer:

```
BEGIN

WHILE TRUE DO

    DISPLAY "Enter a number:"

    INPUT number

    IF number MOD 2 == 0 THEN

        result ← number * number

        DISPLAY "The number is even."

        DISPLAY "Square of the number is: ", result

    ELSE

        result ← number * number * number

        DISPLAY "The number is odd."

        DISPLAY "Cube of the number is: ", result

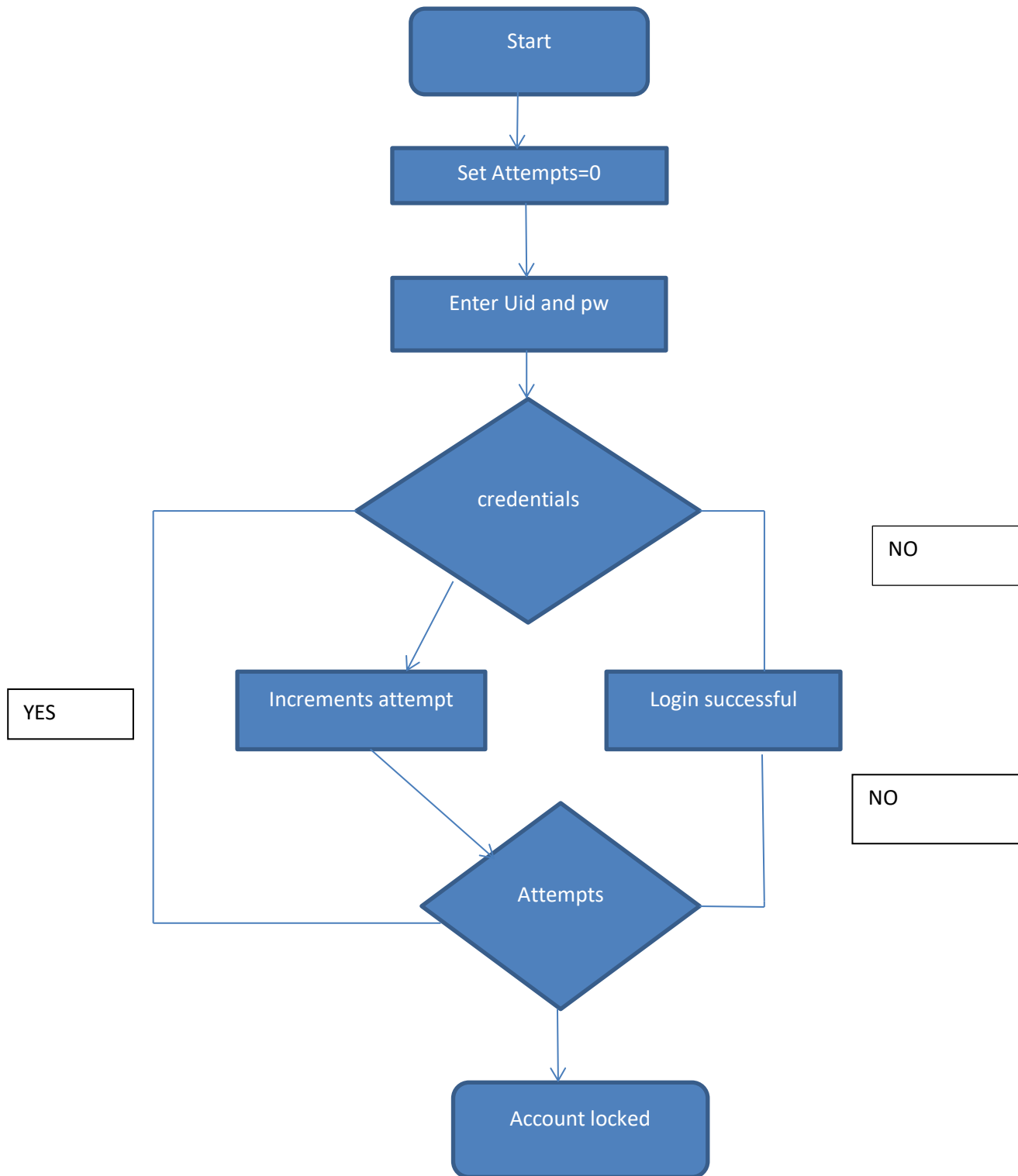
    END IF

END WHILE

END
```

ASSIGNMENT 2: FLOW CHART CREATION

Flowchart Creation - Design a flowchart that outlines the logic for a user login process. It should include conditional paths for successful and unsuccessful login attempts, and a loop that allows a user three attempts before locking the account.



ASSIGNMENT 3

Function Design and Modularization - Create a document that describes the design of two modular functions: one that returns the factorial of a number, and another that calculates the nth Fibonacci number. Include pseudocode and a brief explanation of how modularity in programming helps with code reuse and organization.

Pseudocode - Factorial Function

Function factorial(n)

If $n == 0$ or $n == 1$ then

Return 1

Else

Return $n * \text{factorial}(n - 1)$

End Function

Pseudocode - Fibonacci Function

Function fibonacci(n)

If $n == 0$ then

Return 0

Else if $n == 1$ then

Return 1

Else

Return $\text{fibonacci}(n - 1) + \text{fibonacci}(n - 2)$

End Function

EXPLANATION FOR THIS PROGRAM

This makes your code:

Reusable – you don't have to write the same code again and again.

Easier to fix – if something breaks, you can just look at that small part.

Neater and clearer – your program is easier to read and understand.

Great for teamwork – different people can work on different parts at the same time.