Lab Sheet 06

① What is a recursive method. Briefly explain.

A recursive method is a method that calls itself. There are two key requirements in order to make sure that the recursion is successful.

② What is identified as an iteration.

An iteration is a single repetition of a process. In computer programming, an iteration is a single pass through a loop. Iterations are often used to perform a ~~taks~~ task a specified number of times, or until a certain condition is met.

For example, the following code iterates through a list of numbers and prints each number.

```
numbers = [1, 2, 3, 4, 5]

for number in numbers:
    print(number)
```

SANVIN

examples of iterations :

* A washing machine going through its cycle
* A computer program rendering a frame of animation
* A mathematical algorithm converging on a solution

03. What is Factorial and Fibonacci . Show how they can be can used both as recursive and iterative.

Factorial and Fibonacci are mathematical concept frequently used in computer science and programming. Let's explore what they are and how they can be implemented both recursively

$n! = n * (n-1) * (n-2) * \ldots * 3 * 2 * 1$

Factorial - Recursive Implementation

```
def factorial_recursive(n):
    if n == 0 or n == 1:
        return 1
    else:
        return n * factorial_recursive(n-1)
```

Factorial - Iterative Implementation

```
def factorial_iterative(n):
    result = 1
    for i in range(1, n+1):
        result *= i
    return result
```

WEB CA SAD C#

Date No

→ 0, 1, 1, 2, 3, 5, 8, 13, 21, ...

Fibonacci - Recursive Implementation

```
def fibonacci_recursive(n):
    if n<=0:
        return 0
    elseif n==1:
        return 1
    else:
        return fibonacci_recursive(n-1) + fibonacci_
                                          recursive(n-2)
```

Fibonacci - Iterative Implementation

```
def fibonacci_iterative(n):
    if n<=0:
        return 0
    elseif n==1:
        return 1
    fib_prev, fib_curr = 0,1
    for_in range(2,n+1):
        fib_next = fib_prev + fib_curr
        fib_prev, fib_curr = fib_curr, fib_next
    return fib_curr
```