Tutorial 08

Question

1. What is a recursive method. Briefly explain.

2. What is identified as an iteration. Briefly explain.

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

Answers

1. **Recursive method** is a method that calls itself. This means that the method can be broken down into smaller and smaller subproblems, each of which is solved using the same method. This can be a very efficient way to solve problems, but it can also be difficult to understand and debug.
2. **Iteration** is a technique for solving problems that repeats a set of instructions until a specific condition is met. This is a very common way to solve problems, and it is often easier to understand and debug than recursion.
3. **Factorial** and **Fibonacci** are two mathematical functions that can be expressed recursively or iteratively.

* **Factorial** is the product of all the positive integers less than or equal to a given number. For example, 5! = 120, which is the product of 1, 2, 3, 4, and 5.

def factorial(n):

if n == 0:

return 1

else:

return n \* factorial(n - 1)

The iterative equivalent of the factorial function.

def factorial(n):

result = 1

for i in range(1, n + 1):

result \*= i

return result

* **Fibonacci** is a sequence of numbers where each number is the sum of the two previous numbers. The first two numbers in the Fibonacci sequence are 0 and 1, and the sequence continues as follows: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, ...

def fibonacci(n):

if n == 0:

return 0

elif n == 1:

return 1

else:

return fibonacci(n - 1) + fibonacci(n - 2)

The iterative equivalent of the Fibonacci function.

def fibonacci(n):

a, b = 0, 1

for i in range(1, n + 1):

a, b = b, a + b

return b

Those code are written by Python language.