What is Recursion?

Worksheet

Recursion is when a function calls itself to break a problem into smaller identical subproblems until it reaches a base case that can be answered directly, then the results combine back up.

Questions

1. What is the base case in a recursive function?

A) The first call made
B) The condition that stops further recursive calls
C) The slowest part of the function
D) A syntax error

2. What happens if a recursive function has no base case?

A) It runs faster
B) It causes infinite recursion and a stack overflow
C) It automatically stops after 10 calls
D) Nothing, it works fine

3. In factorial(4) = 4 factorial(3), what is factorial(3) called?

A) The base case
B) The recursive call
C) The return statement
D) An infinite loop

4. What is F(4) if F(n) = F(n-1) + F(n-2), F(0)=0, F(1)=1?

A) 2
B) 3
C) 5
D) 8

5. Compute factorial(5) recursively.

6. Sum the list [3, 5, 2] using recursion.

7. Find Fibonacci(5) using the recursive definition F(n) = F(n-1) + F(n-2), F(0)=0, F(1)=1.

8. Define: What is recursion?

9. Define: What are the two essential parts of a recursive function?

10. Define: What happens without a base case?

Answer Key

1. B) The condition that stops further recursive calls - The base case is the simplest input that can be answered directly without another recursive call, stopping the chain.

2. B) It causes infinite recursion and a stack overflow - Without a base case the function never stops calling itself, exhausting the call stack.

3. B) The recursive call - factorial(3) is the recursive call - the function invoking itself on a smaller input.

4. B) 3 - F(2)=1, F(3)=2, F(4)=F(3)+F(2)=2+1=3.

5. factorial(5) = 5 factorial(4) factorial(4) = 4 factorial(3) factorial(3) = 3 factorial(2) factorial(2) = 2 factorial(1) factorial(1) = 1 factorial(0) = 1 1 = 1 Unwind: 21=2, 32=6, 46=24, 524=120 factorial(5) = 120

6. sum([3,5,2]) = 3 + sum([5,2]) sum([5,2]) = 5 + sum([2]) sum([2]) = 2 + sum([]) = 2 + 0 = 2 Unwind: 5+2=7, 3+7=10 Result: 10

7. F(2) = F(1)+F(0) = 1+0 = 1 F(3) = F(2)+F(1) = 1+1 = 2 F(4) = F(3)+F(2) = 2+1 = 3 F(5) = F(4)+F(3) = 3+2 = 5 Result: F(5) = 5

8. A technique where a function calls itself to solve smaller instances of the same problem.

9. A base case (stops the recursion) and a recursive case (calls itself with a smaller input).

10. Infinite recursion, which eventually crashes with a 'stack overflow' error.