Understanding Recursion

Cracking Coding Interview @ Ostad
- Partharaj Deb -

Introduction to Recursion

A recursive function is a function that calls itself in order to solve a problem.

- Importance in solving problems with repetitive structures.
- The process involves breaking down a problem into smaller, more manageable sub-problems.

Anatomy of a Recursive Function

Base case:

- Definition and purpose.
- Why it's crucial for termination.

Recursive case:

- The part of the function that calls itself.
- The breakdown of a complex problem into simpler subproblems.

Recursion VS Iteration

Recursion	Iteration
Terminates when the base case becomes true.	Terminates when the condition becomes false.
Used with functions.	Used with loops.
Every recursive call needs extra space in the stack memory.	Iteration does not require any extra space.
Smaller code size.	Larger code size.

Visualizing Recursion

Types of Recursion

1. Direct Recursion
    a. Linear Recursion
        i. Tail Recursion
        ii. Non-Tail Recursion / Head Recursion
    b. Tree Recursion
        i. Nested Recursion
2. Indirect Recursion
    a. Mutual Recursion

Example

```
F(n):
    if n == 0: return
    print(n)
    F(n-1)
```

Example: Factorial

```
F(n) = 1           //when n = 0 or 1
     = n x F(n-1)  //when n > 1
```

Example: Fibonacci Sequence

```
Fib(n) = 0                    // when n = 1

       = 1                    // when n = 2

       = Fib(n-1) + Fib(n-2)  // when n > 2
```

```
// 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765, 10946...
```

Advantages of Recursion

- Improves Code Readability and Simplicity
- Improves reusability and reduces redundancy
- Encourages Abstraction
- Solvs problems with inherent recursive structures.
  - Elegant Solutions to Tree and Graph Problems
  - Divide and Conquer Strategy
  - Dynamic Problem-Solving

Challenges of Recursion

- Stack overflow: understanding and avoiding infinite recursion.
- Performance considerations compared to iterative solutions.

Q & A