**1. Exponential Recursion**

**Definition**

Exponential recursion occurs when each recursive call spawns multiple new recursive calls, leading to exponential growth in the number of function calls.

**Example Implementation: Tower of Hanoi**

def hanoi(n, source, auxiliary, target):

if n > 0:

*# Move n-1 disks from source to auxiliary*

hanoi(n - 1, source, target, auxiliary)

*# Move the largest disk*

print(f"Move disk {n} from {source} to {target}")

*# Move n-1 disks from auxiliary to target*

hanoi(n - 1, auxiliary, source, target)

**Growth Pattern**

Copy

n = 1: 1 move

n = 2: 3 moves

n = 3: 7 moves

n = 4: 15 moves

Pattern: 2^n - 1 moves