Set 1: Recursion with Statement *After* Recursive Calls (Unwinding Phase)

Focus: Understanding how the control flows after the base case is hit.

MCQ 1: Basic Print After Recursion

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

show(3)
```

Output?
A) `3 2 1`
B) `1 2 3`
C) `0 1 2`
D) `0 2 3`

MCQ 2: Two Recursive Calls, Post Statement

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

recur(2)
```

Output?
A) `1 1 2`
B) `2 1 1`
C) `1 2 2`
D) `1 2 1`

MCQ 3: Reverse Print with Array

```
arr = [4, 5, 6]
function reversePrint(i):
    if i == length(arr):
        return
    reversePrint(i + 1)
    print(arr[i])

reversePrint(0)
```

Output?
A) `4 5 6`
B) `6 5 4`

C) `6 4 5`
D) `5 6 4`

MCQ 4: Factorial with Print

```
function fact(n):
    if n == 1:
        return 1
    result = n * fact(n - 1)
    print(result)
    return result

fact(4)
```

Output?
A) `4 3 2`
B) `2 6 24`
C) `24 6 2`
D) `2 3 4`

MCQ 5: Sum Accumulator

```
function sum(n, total):
    if n == 0:
        print(total)
        return
    sum(n - 1, total + n)
    print(n)

sum(3, 0)
```

Output?
A) `6 3 2 1`
B) `0 1 2 3`
C) `3 2 1 6`
D) `6 1 2 3`