

Chương 3: Đệ quy (Recursion)

1. Khái niệm đệ quy

- **Đệ quy** là kỹ thuật trong đó một hàm gọi lại chính nó để giải quyết một bài toán.
- Một bài toán được giải bằng cách chia nhỏ nó thành các bài toán con tương tự.

Cấu trúc cơ bản của đệ quy gồm 2 phần:

1. **Điều kiện dừng (base case):** Xác định khi nào hàm không gọi lại chính nó nữa.
2. **Lời gọi đệ quy:** Gọi lại chính hàm đó với một đầu vào nhỏ hơn hoặc đơn giản hơn.

Ví dụ đơn giản - Tính giai thừa $n!$

```
int factorial(int n) {  
    if (n == 0 || n == 1) return 1; // Điều kiện dừng  
    return n * factorial(n - 1); // Lời gọi đệ quy  
}
```

2. Đệ quy tuyến tính (Linear Recursion)

- Là loại đệ quy mà mỗi lời gọi hàm chỉ gọi lại **một lần duy nhất** hàm chính nó.

Ví dụ: Tính tổng các phần tử của mảng

```
int sum(int arr[], int n) {  
    if (n == 0) return 0;  
    return arr[n-1] + sum(arr, n-1);  
}
```

3. Đệ quy nhị phân (Binary Recursion)

- Là loại đệ quy mà mỗi lời gọi hàm gọi lại **hai lần** chính nó (hai nhánh đệ quy).

Ví dụ: Tính số Fibonacci thứ n

```
int fibonacci(int n) {  
    if (n == 0) return 0;  
    if (n == 1) return 1;  
    return fibonacci(n - 1) + fibonacci(n - 2);  
}
```