

# Phân tích hệ thống bài tập cấu tạo số trong lập trình thi đấu

Từ nền tảng cấu trúc điều khiển đến tư duy thuật toán bậc thấp

Slide Learning C++

Rank 800 - 1000

## Biểu diễn số học

Một số nguyên dương  $n$  có thể được phân rã thành:

$$n = a_k \cdot 10^k + a_{k-1} \cdot 10^{k-1} + \dots + a_1 \cdot 10^1 + a_0 \cdot 10^0$$

- **Phép toán trích xuất:**  $a_0 = n \pmod{10}$ .
- **Phép toán loại bỏ:**  $n = n/10$ .
- **Cấu trúc:** Sử dụng `while` để tách chữ số khi chưa biết trước độ dài.

# 1. Sum of Round Numbers (CF 1352A)

**Link:** <https://codeforces.com/problemset/problem/1352/A>

## Đề bài

Phân tích  $n$  ( $n \leq 10^4$ ) thành tổng ít nhất các "số tròn" (dạng  $d \cdot 10^k$ ).

## Hướng giải

- Sử dụng vòng lặp `while` để tách từng chữ số.
- Duyệt qua các vị trí  $i$  từ hàng đơn vị.
- Nếu chữ số  $a_i > 0$ , lưu giá trị  $a_i \cdot 10^i$  vào mảng kết quả.

## 2. Nearest Interesting Number (CF 1183A)

**Link:** <https://codeforces.com/problemset/problem/1183/A>

### Đề bài

Tìm số thú vị nhỏ nhất  $a \geq n$  sao cho tổng các chữ số của nó chia hết cho 4.

### Hướng giải

- Bắt đầu vòng lặp `while(true)` từ giá trị  $n$ .
- **Bước 1:** Dùng vòng lặp con tính tổng các chữ số của số hiện tại.
- **Bước 2:** Dùng `if` để kiểm tra `sum % 4 == 0`.
- Nếu thỏa mãn thì dừng và in kết quả.

### 3. Digits Sum (CF 1553A)

**Link:** <https://codeforces.com/problemset/problem/1553A>

#### Đề bài

Đếm số nguyên  $x \leq n$  sao cho tổng chữ số của  $x + 1$  nhỏ hơn tổng chữ số của  $x$ .

#### Phân tích quy luật

- Quy luật: Tổng chữ số chỉ giảm khi có hiện tượng nhớ (ví dụ từ 9 lên 10).
- Chỉ các số kết thúc bằng chữ số 9 mới thỏa mãn điều kiện này.
- **Công thức:**  $(n + 1)/10$ .

## 4. Distinct Digits (CF 1228A)

**Link:** <https://codeforces.com/problemset/problem/1228/A>

### Đề bài

Tìm một số  $x$  trong đoạn  $[l, r]$  có các chữ số đôi một khác nhau.

### Hướng giải

- Duyệt for từ  $l$  đến  $r$ .
- Với mỗi số, dùng while để tách từng chữ số.
- Sử dụng mảng đánh dấu bool `appeared[10]` để kiểm tra xem chữ số đã xuất hiện chưa.
- Nếu tìm thấy số thỏa mãn, in ra và kết thúc chương trình.

## 5. Ordinary Numbers (CF 1520B)

**Link:** <https://codeforces.com/problemset/problem/1520/B>

### Đề bài

Đếm số lượng số chỉ gồm một loại chữ số (1, 22, 333,...) trong đoạn  $[1, n]$ .

### Hướng giải

- Thay vì duyệt đến  $10^9$ , ta chủ động sinh ra các số này.
- Dùng 2 vòng lặp lồng nhau:
  - Vòng ngoài cho chữ số từ 1 đến 9.
  - Vòng trong tạo số dạng  $d, dd, ddd, \dots$
- Đếm các số sinh ra mà  $\leq n$ .

## 6. Div. 7 (CF 1633A)

**Link:** <https://codeforces.com/problemset/problem/1633/A>

### Đề bài

Thay đổi ít nhất chữ số của  $n$  để được số mới chia hết cho 7.

### Hướng giải

- Nếu  $n \pmod{7} == 0$ , giữ nguyên.
- Nếu không, thử thay đổi chữ số hàng đơn vị.
- Duyệt for từ 0 đến 9 để thay thế chữ số cuối.
- Kiểm tra số mới, nếu chia hết cho 7 thì dừng lại.
- **Lưu ý:** Đảm bảo số mới không làm thay đổi số lượng chữ số ban đầu.

- **Kỹ năng tách số:** Là xương sống của các bài toán cấu tạo số.
- **Độ phức tạp:**  $O(\log_{10} n)$  cho mỗi lần xử lý chữ số, cực kỳ tối ưu.
- **Tiếp cận:**

- **Kỹ năng tách số:** Là xương sống của các bài toán cấu tạo số.
- **Độ phức tạp:**  $O(\log_{10} n)$  cho mỗi lần xử lý chữ số, cực kỳ tối ưu.
- **Tiếp cận:**
- Luôn kiểm tra các trường hợp biên (số 0, số có toàn chữ số 9).

- **Kỹ năng tách số:** Là xương sống của các bài toán cấu tạo số.
- **Độ phức tạp:**  $O(\log_{10} n)$  cho mỗi lần xử lý chữ số, cực kỳ tối ưu.
- **Tiếp cận:**
  - Luôn kiểm tra các trường hợp biên (số 0, số có toàn chữ số 9).
  - Cân nhắc giữa việc duyệt số và việc sinh số.