

Codeforces 1884A - Simple Design

Algorithmic Coach - Learning How to Learn

Slide Learning C++

Ngày 21 tháng 1 năm 2026

Chào bạn!

Tôi đã sẵn sàng nhập vai **Algorithmic Coach** để cùng bạn chinh phục bài toán này theo đúng tinh thần "Learning How to Learn".

- Chúng ta sẽ không vội vàng viết code.
- Hãy bắt đầu bằng việc "phẫu thuật" bài toán để hiểu rõ bản chất.

Bước 1: Phân thuật đề bài (Deconstruct)

- **Input:** Hai số nguyên x (điểm xuất phát) và k (mục tiêu về tổng các chữ số).
- **Yêu cầu:** Tìm số y nhỏ nhất sao cho:
 - 1 $y \geq x$.
 - 2 Tổng các chữ số của y phải chia hết cho k .

Ấn dụ

Tưởng tượng bạn đứng ở bậc thang thứ x . Bạn bước lên từng bậc cho đến khi tìm được bậc "may mắn" mà tổng các chữ số ghi trên đó chia hết cho k .

- **Chunk 1:** Cách tính tổng các chữ số.
- **Chunk 2:** Chiến thuật tìm kiếm.
- **Chunk 3:** Ước lượng giới hạn.

Chunk 1: Cách "mổ bụng" một số để tính tổng

- Ví dụ: Số 123 có tổng là $1 + 2 + 3 = 6$.

Bẫy tư duy

Đừng coi số đó là một khối liền mạch. Hãy coi nó là một chuỗi các "ngăn kéo", mỗi ngăn đựng một chữ số từ 0 đến 9.

Thử thách tư duy 1

Nếu có $x = 432$ và $k = 10$:

- 1 Tổng các chữ số hiện tại của x là bao nhiêu?
- 2 Tổng đó có chia hết cho k không?

Chunk 1: Cách "mổ bụng" một số để tính tổng

- Ví dụ: Số 123 có tổng là $1 + 2 + 3 = 6$.

Bẫy tư duy

Đừng coi số đó là một khối liền mạch. Hãy coi nó là một chuỗi các "ngăn kéo", mỗi ngăn đựng một chữ số từ 0 đến 9.

Thử thách tư duy 1

Nếu có $x = 432$ và $k = 10$:

- 1 Tổng các chữ số hiện tại của x là bao nhiêu?
- 2 Tổng đó có chia hết cho k không?

Đáp án: Tổng là $4 + 3 + 2 = 9$. Vì 9 không chia hết cho 10, x không phải số "may mắn".

Chunk 2: Chiến thuật tìm kiếm (Brute Force)

- Đề bài yêu cầu tìm y **nhỏ nhất** mà $y \geq x$.
- **Chiến thuật:** Thử với chính x . Nếu không thỏa mãn, tăng x lên 1 đơn vị và kiểm tra tiếp.

Ẩn dụ: Cửa hàng mở cửa

Đi bộ tuần tự trên phố, tạt vào từng cửa hàng kế tiếp cho đến khi thấy biển "Open". Cửa hàng đầu tiên thấy chắc chắn là gần nhất.

Bẫy logic

Đừng cố dùng toán học phức tạp để "tính" ngay ra y . Brute Force ở đây cực kỳ hiệu quả vì khoảng cách giữa các số thỏa mãn thường rất ngắn.

Thử thách tư duy 2

Ví dụ $x = 432$ và $k = 10$

- 1 Sau số 432, số tiếp theo bạn kiểm tra là số nào?
- 2 Số đó có tổng các chữ số bao nhiêu? Có chia hết cho k không?

Thử thách tư duy 2

Ví dụ $x = 432$ và $k = 10$

- 1 Sau số 432, số tiếp theo bạn kiểm tra là số nào?
- 2 Số đó có tổng các chữ số bao nhiêu? Có chia hết cho k không?
 - Tiếp theo là 433 (tổng 10).
 - Vì 10 chia hết cho $k = 10$, nên 433 là kết quả nhỏ nhất!

Chunk 3: Ước lượng giới hạn

- **Câu hỏi:** Liệu máy tính có chạy kịp nếu x rất lớn?
- **Phân tích:** Với $k \leq 40$, trong một khoảng ngắn các số liên tiếp (thường không quá 40 số), chắc chắn sẽ có một số thỏa mãn.
- Vòng lặp sẽ không bao giờ chạy quá lâu.

Tổng kết thuật toán (Pseudocode)

```
1 1. Nhập x và k.  
2 2. Tạo một vòng lặp bắt đầu từ y = x:  
3   A. Tính tổng các chữ số của y (dùng % 10).  
4   B. Kiểm tra xem tổng đó có chia hết cho k hay không.  
5   C. Nếu CÓ:  
6       In ra y và dừng vòng lặp.  
7   Nếu KHÔNG:  
8       Tăng y lên 1 và quay lại Bước A.
```

Listing 1: Mã giả thuật toán

Thử thách cuối cùng

Đóng vai máy tính

Nếu $x = 19$ và $k = 5$. Hãy chạy từng bước vòng lặp và cho biết **số y đầu tiên** thỏa mãn là số nào?

Thử thách cuối cùng

Đóng vai máy tính

Nếu $x = 19$ và $k = 5$. Hãy chạy từng bước vòng lặp và cho biết **số y đầu tiên** thỏa mãn là số nào?

- $19 \rightarrow$ tổng 10 (chia hết cho 5).
- **Kết quả:** 19. (Đừng quên kiểm tra chính số x đầu tiên nhé!)