

Huấn luyện Tư duy Thuật toán

Codeforces 1633A - Div. 7

Slide Learning CPP

2026

Chào mừng bạn!

Triết lý Learning How to Learn

Chào mừng bạn đến với lộ trình "mổ xẻ" bài toán chuyên sâu. Chúng ta sẽ không chỉ học Code, chúng ta học cách tư duy để giải quyết vấn đề.

Bước 1: Phẫu thuật đề bài (Briefing)

- **Đầu vào:** Một số nguyên n .
- **Nhiệm vụ:** Tìm số m sao cho $m \vdots 7$ và m khác biệt với n ở ít chữ số nhất.
- **Ràng buộc:**
 - Không thay đổi số lượng chữ số (ví dụ: n có 2 chữ số thì m phải có 2 chữ số).
 - Không được có số 0 ở đầu (trừ khi chính nó là số 0).

Lộ trình tư duy

1. Kiểm tra trạng thái "Lý tưởng".
2. Chiến thuật "Sửa chữa tối thiểu".
3. Tìm quy luật trên vạch số.

Chunk 1: Trạng thái "Lý tưởng" & Khoảng cách số

Hãy tưởng tượng số 7 giống như một cái trạm xe buýt xuất hiện đều đặn: 7, 14, 21, 28...

Bẫy logic

Nếu số n đã chia hết cho 7, mục tiêu của chúng ta là **thay đổi ít nhất có thể** (tức là thay đổi 0 chữ số).

Chunk 1: Trạng thái "Lý tưởng" & Khoảng cách số

Hãy tưởng tượng số 7 giống như một cái trạm xe buýt xuất hiện đều đặn: 7, 14, 21, 28...

Bẫy logic

Nếu số n đã chia hết cho 7, mục tiêu của chúng ta là **thay đổi ít nhất có thể** (tức là thay đổi 0 chữ số).

Thử thách tư duy

Nếu tôi đưa cho bạn số $n = 42$. Bạn có cần thay đổi chữ số nào không? Kết quả trả về là bao nhiêu?

Chunk 1: Trạng thái "Lý tưởng" & Khoảng cách số

Hãy tưởng tượng số 7 giống như một cái trạm xe buýt xuất hiện đều đặn: 7, 14, 21, 28...

Bẫy logic

Nếu số n đã chia hết cho 7, mục tiêu của chúng ta là **thay đổi ít nhất có thể** (tức là thay đổi 0 chữ số).

Thử thách tư duy

Nếu tôi đưa cho bạn số $n = 42$. Bạn có cần thay đổi chữ số nào không? Kết quả trả về là bao nhiêu?

Trả lời: Không cần đổi. Kết quả là 42 (vì $42:7$).

Trường hợp "Không lý tưởng"

Xét $n = 48$. Số này không chia hết cho 7. Các trạm xe buýt gần đó:

- Số nhỏ hơn gần nhất: 42 (thay 8 thành 2 - đổi 1 chữ số).
- Số lớn hơn gần nhất: 49 (thay 8 thành 9 - đổi 1 chữ số).

Vùng an toàn

Khoảng cách giữa các số chia hết cho 7 chỉ là **7 đơn vị**. Điều này cực kỳ quan trọng cho chiến thuật tiếp theo.

Chunk 2: Chiến thuật "Giữ nguyên hàng chục"

- Mục tiêu: Chỉ tìm một chữ số hàng đơn vị mới để $m \div 7$.
- Ví dụ với $n = 48$: Thử các số từ 40 đến 49.

Thử thách tư duy

Tại sao trong một dãy 10 số liên tiếp (ví dụ từ 40 đến 49), chúng ta **luôn luôn** tìm được ít nhất một số chia hết cho 7?

Chunk 2: Chiến thuật "Giữ nguyên hàng chục"

- Mục tiêu: Chỉ tìm một chữ số hàng đơn vị mới để $m \div 7$.
- Ví dụ với $n = 48$: Thử các số từ 40 đến 49.

Thử thách tư duy

Tại sao trong một dãy 10 số liên tiếp (ví dụ từ 40 đến 49), chúng ta **luôn luôn** tìm được ít nhất một số chia hết cho 7?

Gợi ý: Khoảng cách giữa các số chia hết cho 7 là 7 đơn vị, nhỏ hơn độ dài dãy (10 đơn vị).

Chunk 3: Xử lý "Bẫy" hàng đơn vị

Rủi ro

Nếu ta thay đổi chữ số bằng cách cộng/trừ đơn thuần, có thể vô tình làm thay đổi hàng chục (ví dụ: $48 + 3 = 51$).

Chiến thuật an toàn

Chỉ tìm kiếm trong phạm vi các số có cùng chữ số hàng chục với n .

- 1 $n = 48 \rightarrow$ Quét dãy $[40, 49]$.
- 2 Tìm thấy 42 hoặc 49. Cả hai đều chỉ khác n đúng 1 chữ số.

Chunk 3: Xử lý "Bẫy" hàng đơn vị

Rủi ro

Nếu ta thay đổi chữ số bằng cách cộng/trừ đơn thuần, có thể vô tình làm thay đổi hàng chục (ví dụ: $48 + 3 = 51$).

Chiến thuật an toàn

Chỉ tìm kiếm trong phạm vi các số có cùng chữ số hàng chục với n .

- 1 $n = 48 \rightarrow$ Quét dãy $[40, 49]$.
- 2 Tìm thấy 42 hoặc 49. Cả hai đều chỉ khác n đúng 1 chữ số.

Thử thách cuối

Nếu $n = 882$, bạn sẽ "quét" qua dãy số nào?

Chunk 3: Xử lý "Bẫy" hàng đơn vị

Rủi ro

Nếu ta thay đổi chữ số bằng cách cộng/trừ đơn thuần, có thể vô tình làm thay đổi hàng chục (ví dụ: $48 + 3 = 51$).

Chiến thuật an toàn

Chỉ tìm kiếm trong phạm vi các số có cùng chữ số hàng chục với n .

- 1 $n = 48 \rightarrow$ Quét dãy $[40, 49]$.
- 2 Tìm thấy 42 hoặc 49. Cả hai đều chỉ khác n đúng 1 chữ số.

Thử thách cuối

Nếu $n = 882$, bạn sẽ "quét" qua dãy số nào?

Đáp án: Dãy từ 880 đến 889.

Chien luoc tong quat (Grand Strategy)

```
1 1. Neu n chia het cho 7:  
2   In ra n va dung lai.  
3 2. Neu khong:  
4   - Tim n_base = (n / 10) * 10 (vi du 48 -> 40)  
5   - Lap i tu 0 den 9:  
6     - Neu (n_base + i) chia het cho 7:  
7       In ra (n_base + i) va dung lai.  
8
```

Listing 1: Ma gia thuat toan

Uu diem

Đảm bảo số chữ số không đổi và số lượng thay đổi luôn là tối ưu (0 hoặc 1 chữ số).

Thử thách thực thi

Tới lượt bạn!

Bạn hãy thử viết code cho bài này bằng ngôn ngữ bạn thích (C++, Python, Java...).

Bạn muốn tự tay thực hiện hay muốn xem mã nguồn mẫu từ Coach?