

Bài toán 1107/B - Digital Root

Algorithmic Coach - Triết lý Learning How to Learn

Slide Learning C++

Ngày 21 tháng 1 năm 2026

Bước 1: Tiếp nhận & Phẫu thuật (Briefing)

Tóm tắt đề bài "Ngôn ngữ con người"

Digital Root (Căn nguyên số) là kết quả cuối cùng sau khi cộng dồn các chữ số của nó cho đến khi chỉ còn lại một chữ số duy nhất (từ 1 đến 9).

Ví dụ

$5 + 8 = 13 \rightarrow 1 + 3 = 4$. Vậy Digital Root của 58 là 4.

Yêu cầu

Tìm số nguyên dương lớn thứ k theo thứ tự tăng dần có Digital Root bằng đúng giá trị x cho trước.

Chúng ta sẽ chia bài toán thành 3 khối kiến thức (Chunks):

- **Chunk 1:** Khám phá quy luật của Digital Root.
- **Chunk 2:** Tìm công thức liên hệ giữa một số và Digital Root.
- **Chunk 3:** Xây dựng chiến thuật và xử lý kiểu dữ liệu.

Chunk 1: Quy luật của những vòng lặp

Hãy tưởng tượng Digital Root giống như một chiếc đồng hồ có 9 số (từ 1 đến 9).

Thử thách tư duy

Hãy tính Digital Root của các số sau:

- Số **10** có Digital Root là ?

Chunk 1: Quy luật của những vòng lặp

Hãy tưởng tượng Digital Root giống như một chiếc đồng hồ có 9 số (từ 1 đến 9).

Thử thách tư duy

Hãy tính Digital Root của các số sau:

- Số **10** có Digital Root là: ? **1**
- Số **11** có Digital Root là: ?

Chunk 1: Quy luật của những vòng lặp

Hãy tưởng tượng Digital Root giống như một chiếc đồng hồ có 9 số (từ 1 đến 9).

Thử thách tư duy

Hãy tính Digital Root của các số sau:

- Số **10** có Digital Root là: ? **1**
- Số **11** có Digital Root là: ? **2**
- Số **18** có Digital Root là: ?

Chunk 1: Quy luật của những vòng lặp

Hãy tưởng tượng Digital Root giống như một chiếc đồng hồ có 9 số (từ 1 đến 9).

Thử thách tư duy

Hãy tính Digital Root của các số sau:

- Số **10** có Digital Root là: ? **1**
- Số **11** có Digital Root là: ? **2**
- Số **18** có Digital Root là: ? **9**
- Số **19** có Digital Root là: ?

Chunk 1: Quy luật của những vòng lặp

Hãy tưởng tượng Digital Root giống như một chiếc đồng hồ có 9 số (từ 1 đến 9).

Thử thách tư duy

Hãy tính Digital Root của các số sau:

- Số **10** có Digital Root là: ? **1**
- Số **11** có Digital Root là: ? **2**
- Số **18** có Digital Root là: ? **9**
- Số **19** có Digital Root là: ? **1**

Chunk 1: Quan sát quan trọng

Kết quả

Cứ sau **9 đơn vị**, Digital Root lại lặp lại một lần:

- Số 1 $\rightarrow DR = 1$
- Số 10 ($1 + 9$) $\rightarrow DR = 1$
- Số 19 ($10 + 9$) $\rightarrow DR = 1$

Nhận xét: Các số có cùng Digital Root x tạo thành một cấp số cộng với công sai là **9**. Số đầu tiên nhỏ nhất luôn là x .

Chunk 2: Tìm công thức tổng quát

Tìm số thứ k trong dãy các số có Digital Root bằng x :

- ❶ Số thứ 1: x
- ❷ Số thứ 2: $x + 9$
- ❸ Số thứ 3: $x + 18 = x + 2 \cdot 9$
- ❹ Số thứ 4: $x + 27 = x + 3 \cdot 9$

Chunk 2: Tìm công thức tổng quát

Tìm số thứ k trong dãy các số có Digital Root bằng x :

- ❶ Số thứ 1: x
- ❷ Số thứ 2: $x + 9$
- ❸ Số thứ 3: $x + 18 = x + 2 \cdot 9$
- ❹ Số thứ 4: $x + 27 = x + 3 \cdot 9$

Thử thách tư duy

Dựa vào mối liên hệ giữa số thứ tự k và hệ số nhân với 9, công thức tổng quát là gì?

Chunk 2: Tìm công thức tổng quát

Tìm số thứ k trong dãy các số có Digital Root bằng x :

- ① Số thứ 1: x
- ② Số thứ 2: $x + 9$
- ③ Số thứ 3: $x + 18 = x + 2 \cdot 9$
- ④ Số thứ 4: $x + 27 = x + 3 \cdot 9$

Thử thách tư duy

Dựa vào mối liên hệ giữa số thứ tự k và hệ số nhân với 9, công thức tổng quát là gì?

$$\text{Kết quả} = x + (k - 1) \cdot 9$$

Chunk 3: Xử lý bẫy về kiểu dữ liệu

Giới hạn đề bài

- Số bộ test $n \leq 10^3$.
- Thứ tự $k \leq 10^{12}$.
- Giá trị $x \in [1, 9]$.

Chunk 3: Xử lý bẫy về kiểu dữ liệu

Giới hạn đề bài

- Số bộ test $n \leq 10^3$.
- Thứ tự $k \leq 10^{12}$.
- Giá trị $x \in [1, 9]$.

Phân tích

Nếu $k = 10^{12}$ và $x = 9$, kết quả sẽ xấp xỉ $9 \cdot 10^{12}$. Kiểu `int` trong C++ chỉ lưu được đến khoảng $2 \cdot 10^9$.

Chunk 3: Xử lý bẫy về kiểu dữ liệu

Giới hạn đề bài

- Số bộ test $n \leq 10^3$.
- Thứ tự $k \leq 10^{12}$.
- Giá trị $x \in [1, 9]$.

Phân tích

Nếu $k = 10^{12}$ và $x = 9$, kết quả sẽ xấp xỉ $9 \cdot 10^{12}$. Kiểu `int` trong C++ chỉ lưu được đến khoảng $2 \cdot 10^9$.

Giải pháp: Sử dụng kiểu dữ liệu `long long` để tránh lỗi tràn số (overflow).

Chunk 4: Tổng kết & Mã giả

Lưu đồ thuật toán

- ❶ Đọc số lượng bộ test n .
- ❷ Lặp n lần:
 - Đọc k và x .
 - Áp dụng công thức $x + (k - 1) \cdot 9$.
 - In kết quả.

```
1 Nhap n (so luong bo test)
2 Lap n lan:
3     Nhap k, x
4     Ket_qua = x + (k - 1) * 9
5     In Ket_qua
```

Listing 1: Pseudocode

Chúc các bạn học tập tốt!

Bạn có muốn tôi hỗ trợ viết mã nguồn hoàn chỉnh bằng C++ hay Python không?