

CHUYÊN ĐỀ 1. TÌM SỐ TỰ NHIÊN

Tác giả: Phan Thành Hưng (PTNK, ĐHQG TP. HCM)

I. Giới thiệu

Chuyên đề này sẽ đề cập tới cách giải của bài toán sau:

“Tìm số tự nhiên X nhỏ nhất có m chữ số thỏa X chia a dư b và X chia c dư d ”.

II. Lời giải

Lời giải tổng quát của bài toán như sau:

Ta viết lại như sau:

$$\begin{cases} X = a \cdot q_1 + b \\ X = c \cdot q_2 + d \end{cases}$$

Với $q_1, q_2 \in \mathbb{N}$

$$\Rightarrow a \cdot q_1 + b = c \cdot q_2 + d$$

$$\Rightarrow q_1 = \frac{c \cdot q_2 + d - b}{a}$$

Trước hết, ta kiểm tra xem liệu có tồn tại q_1 nguyên hay không bằng cách đưa về dạng phương trình diophantine:

$$a \cdot q_1 + b = c \cdot q_2 + d$$

$$a \cdot q_1 + -c \cdot q_2 = d - b$$

Đây là một phương trình diophantine tuyến tính (có dạng $ax + by = c$). Phương trình này tồn tại nghiệm x, y nguyên khi và chỉ khi c chia hết cho $\gcd(a, b)$.

$\Rightarrow$ **Bài toán chỉ tồn tại nghiệm khi $d - b$ chia hết cho $\gcd(a, c)$.**

$\Rightarrow$ [!] Tài liệu tham khảo thêm:

https://en.wikipedia.org/wiki/Diophantine_equation#One_equation

Lưu ý: Nếu đọc xong đoạn này chả hiểu gì thì cứ nhớ phần in đậm / gạch chân là đủ.

Lập table duyệt giá trị x từ 0 trở lên cho tới khi tìm được x đầu tiên mà $\frac{c \cdot x + d - b}{a}$ là một số nguyên **không âm** thì dừng lại. Ta đặt $V = c \cdot x + d$, lúc này q là số nguyên không âm nhỏ nhất thỏa mãn điều kiện chia a dư b và chia c dư d .

Ta cần tìm số L nhỏ nhất sao cho $V + L$ vẫn thỏa mãn điều kiện này. Biết rằng,

$$\begin{cases} V \equiv b \pmod{a} \\ V \equiv d \pmod{c} \end{cases}$$

Cần tìm l nhỏ nhất thỏa mãn:
$$\begin{cases} V + L \equiv b \pmod{a} \\ V + L \equiv d \pmod{c} \end{cases}$$

$\Rightarrow L \equiv 0 \pmod{a}$ và $L \equiv 0 \pmod{c}$ và L là nhỏ nhất.

$\Rightarrow L$ là BCNN của a và c hay $L = \text{lcm}(a, c)$.

Do vậy nếu V thỏa mãn điều kiện trên thì $V + L$ cũng thỏa mãn điều kiện trên, dẫn tới $V + L \cdot k$ ($\forall k \in \mathbb{N}$) cũng phải thỏa mãn điều kiện như vậy. Hay X là một số nguyên không âm nhỏ nhất có m chữ số có dạng $V + L \cdot k$.

Số nguyên không âm nhỏ nhất có m chữ số là 10^{m-1} .

Ta có:

$$X = V + L \cdot k \geq 10^{m-1}$$

$$k \geq \frac{10^{m-1} - V}{L}$$

$$k_{\min} = \left\lceil \frac{10^{m-1} - V}{L} \right\rceil$$

Vậy giá trị $k_{\min}$ cần tìm để X đạt nhỏ nhất và có m chữ số là $\left\lceil \frac{10^{m-1} - V}{L} \right\rceil$ ($\lceil x \rceil$ là [hàm ceiling](#)).

Kết quả là $X = V + \text{lcm}(a, c) \times \left\lceil \frac{10^{m-1} - V}{L} \right\rceil$

với V là số nguyên không âm nhỏ nhất thỏa điều kiện chia a dư b và chia c dư d , có thể tìm bằng tính năng table.

III. Tổng quát hóa

Giả sử ta cần tìm số X nhỏ nhất có m chữ số thỏa mãn n điều kiện sau: X chia a_1 dư b_1 , X chia a_2 dư $b_2, \dots$, X chia a_n dư b_n , hay:

$$\begin{cases} X \equiv b_1 \pmod{a_1} \\ X \equiv b_2 \pmod{a_2} \\ \vdots \\ X \equiv b_n \pmod{a_n} \end{cases}$$

Dựa vào bài toán trên, ta đã có lời giải cho 2 điều kiện:

$$\begin{cases} X \equiv b_1 \pmod{a_1} \\ X \equiv b_2 \pmod{a_2} \end{cases} \quad (1)$$

Lúc này, đặt $L = \text{lcm}(a_1, a_2)$ và V là số nguyên không âm nhỏ nhất thỏa (1) thì X có dạng $V + \text{lcm}(a_1, a_2) \cdot k$ (đã giải ở bài toán trên). Từ đó ta nhận thấy một điều đó chính là:

$$X \equiv V \pmod{L}$$

Do đó, ta có thể nhóm hai điều kiện bất kì thành một điều kiện chung, từ đó lần lượt giảm số điều kiện ta cần xử lý xuống cho tới khi còn đúng 2 điều kiện, ta xử lý y hệt bài toán trên. Cụ thể:

$$\begin{cases} X \equiv b_1 \pmod{a_1} \\ X \equiv b_2 \pmod{a_2} \\ X \equiv b_3 \pmod{a_3} \\ \vdots \\ X \equiv b_n \pmod{a_n} \end{cases} \Leftrightarrow \begin{cases} X \equiv V \pmod{L} \\ X \equiv b_3 \pmod{a_3} \\ \vdots \\ X \equiv b_n \pmod{a_n} \end{cases} \Rightarrow \text{còn } n - 1 \text{ điều kiện}$$

Cứ nhóm 2 điều kiện với nhau như vậy, cho tới khi còn 2 điều kiện cuối cùng, ta xử lý y hệt như bài toán trên và tìm ra được đáp án.

IV. Mở rộng: một cách giải “nhANH HƠN”? [NÂNG CAO]

Nhận thấy rằng, việc chạy table rất tốn thời gian và có nhiều trường hợp khi ta phải chạy x lên tới cả trăm, hoặc cả ngàn thì mới tìm ra được nghiệm. Do đó, ta biến đổi bài toán trở về dạng phương trình diophantine tuyến tính như đã làm ở trên:

$$a \cdot q_1 + -c \cdot q_2 = d - b \quad (*)$$

Đặt $g = \gcd(a, c)$. Bài toán có nghiệm khi và chỉ khi $d - b$ chia hết cho g .

Trong đó, ta có thể giải bài toán sau:

$$ax + by = \gcd(a, b)$$

bằng cách sử dụng thuật toán [Euclid mở rộng](#). Do đó, ta biến đổi phương trình (*) thành dạng:

$$a \cdot q'_1 + -c \cdot q'_2 = g$$

Sau khi sử dụng thuật toán Euclid mở rộng, ta nhân cả q'_1 và q'_2 cho $\frac{d-b}{g}$, biến đổi lại và nhận được nghiệm q_1 và q_2 cho phương trình (*) cần giải.

Trên lý thuyết ta có thể tìm ra nghiệm của bài toán chỉ trong khoảng xấp xỉ $\log_{10}(\min(a, c))$ lần chạy, nhanh hơn rất nhiều so với cách làm chạy table từng số một như cũ, nhưng hầu hết các bài toán trong MTCT không cần tối ưu đến mức như vậy vì chỉ cần chạy table tới vài chục số là đã tìm ra kết quả rồi. Ngoài ra, việc cài đặt thuật toán này vào máy tính Casio rất dài dòng và không hề dễ dàng chút nào. Do vậy, cách làm cũ vẫn được ưa chuộng nhiều hơn. Tuy nhiên, nếu bạn đã quen cách này và quen luôn cách cài đặt rồi thì đây là một cách giúp ta tiết kiệm được *một ít* thời gian làm bài.

V. Giới thiệu chương trình Scratch giải dạng bài này tự động

Truy cập chương trình Scratch sau: <https://scratch.mit.edu/projects/1048936510>

Xem cách sử dụng chương trình trong phần **Instructions** và nhập dữ liệu vào mong muốn.

Ví dụ kết quả với bài toán sau:

Tìm số tự nhiên x nhỏ nhất có 10 chữ số, biết x chia cho 17 dư 5, chia cho 29 dư 11 và chia cho 43 dư 25.

Sau khi nhập dữ liệu bài toán vào tương ứng, ta nhận kết quả sau:

Tìm số tự nhiên
by chxc0ny

See inside

Instructions

Sử dụng để tìm số tự nhiên x NHỎ NHẤT có TỐI THIỂU # chữ số biết số dư của x cho nhiều số nguyên dương khác.

Cách sử dụng:

1. Nhập số chữ số TỐI THIỂU của x .
2. Nhập số cần số dư (một cặp a, b nghĩa là x chia a dư b).

Notes and Credits

Tạo bởi: Lớp 9A2

Nếu có lỗi hoặc góp ý, hãy bình luận xuống dưới :)
[ĐỒNG BÌNH LUẬN = BẢO TRÌ]

Phiên bản: 1.1.0

Jul 23, 2024

Copy Link

Thật vậy, **1000000457** là kết quả bài toán.

Fun facts về chương trình này :D

- Nó được tạo ra bởi chính mình vào 1 năm trước khi mình quá chán và không có gì làm cả
- Thuật toán được sử dụng để tìm ra kết quả (chạy table) không phải là thuật toán tốt nhất, vẫn còn cách nhanh hơn là thuật Euclid mở rộng. Tuy nhiên, hồi đó do trình độ còn gà nên mình dùng thuật toán đó, nhưng thuật toán mới là Euclid mở rộng thực ra cài đặt cũng chẳng dễ tí nào ;o)
- Cũng đã hơn 1 năm từ lần cuối mình check lại dự án này rồi nên chương trình vẫn có thể xảy ra lỗi về kết quả ;_o. Nếu có cứ mạnh dạn liên hệ mình để mình sửa lại nhé :D