

Chuyên đề nâng cao: [Cải tiến cách cài đặt] Thuật toán Pollard's rho

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

I. Giới thiệu

Cho bài toán sau:

“Cho một hợp số n , hãy xác định một ước số D của n sao cho $1 < D < n$ ”

II. Thuật toán

a. THUẬT TOÁN CHÍNH (Pollard's rho)

Thuật toán chi tiết đã được mình giải thích và giới thiệu khá kỹ ở bài viết trước. Tuy nhiên mình vẫn sẽ viết lại các bước chính của thuật toán:

- ✚ Khởi tạo một biến x, y , đều mang giá trị ngẫu nhiên ban đầu.
- ✚ Khởi tạo $D = 1$.
- ✚ Lặp cho tới khi $D \neq 1$:
 - $x := f(x)$ // rùa
 - $y := f(f(y))$ // thỏ
 - $D := \text{GCD}(|x - y|, n)$
- ✚ Nếu $D = n \rightarrow$ thử lại hàm f với một hằng số c mới.
- ✚ Nếu $D \neq n$ thì ta đã tìm được D thỏa mãn.

➤ Cài đặt vào CASIO

Ta cần xử lý điều kiện “lặp cho tới khi $D \neq 1$ ”. Ta có thể tạo một biến F cho phép ta quản lý việc “có lặp tiếp hay không?”. Nếu $F = 0$ thì nghĩa là $D \neq 1$ và ta cho dừng vòng lặp. Nhận thấy:

$$\begin{cases} F = 0, & \text{nếu } D - 1 \neq 0 \\ F = 1, & \text{nếu } D - 1 = 0 \end{cases}$$

Tên gọi của những hàm có dạng như vậy là [hàm delta Kronecker](#) trong toán rời rạc.

Vì D là ước chung lớn nhất của hai số nguyên dương nên luôn dương, cụ thể $D \geq 1$. Ta có thể tận dụng điều đó để xử lý điều kiện $D = 1$ một cách dễ dàng mà không dùng tới hàm cos phức tạp như trước.

$$\begin{cases} F = \text{Int}\left(\frac{1}{D}\right) = 0, \Leftrightarrow D > 1, D \in \mathbb{N} \\ F = \text{Int}\left(\frac{1}{D}\right) = 1, \Leftrightarrow D = 1, D \in \mathbb{N} \end{cases} \quad [*]$$

Lúc này, nếu tiếp tục vòng lặp, giá trị x, y sẽ giữ nguyên để chạy tiếp, ta có thể xử lý chúng một cách đơn giản bằng phép gán:

$$\begin{cases} x := (1 - F)x + Fx \\ y := (1 - F)x + Fy \end{cases}$$

Tiếp theo, ta cần xử lý việc dừng thuật toán. Ta dừng thuật toán khi $1 < D < n$, nghĩa là $D \neq 1$ **VÀ** $D \neq n$. Điều đó tương đương với việc không phải ($D = 1$ hoặc $D = n$) xảy ra. Cụ thể: $\neg(D = 1) \wedge \neg(D = n) = \neg(D = 1 \vee D = n)$. Nếu xét chúng là một đại lượng mang giá trị 0/1 thì nghĩa là $(D = 1) + (D = n) = 0 \Leftrightarrow F + (D = n) = 0$. Nhận thấy rằng ta có thể cố tình ép máy tính gây lỗi để báo dừng bằng cách đưa biểu thức trên xuống mẫu số của một phân số để máy báo **Math ERROR**.

Vì D là ước của n (từ ƯCLN) nên ta có thể mô phỏng giá trị boolean của $D = n$ bằng kỹ thuật **[*]** ở trên. Do đó, biểu thức giúp ta dừng vòng lặp là: $0 \div \left(F + \text{Int}\left(\frac{D}{n}\right)\right)$.

Kết hợp những thứ ở trên, ta có một thuật toán hoàn chỉnh cài được vào máy tính Casio fx-580 VN X:

$$x = 0MFD + (1 - F) \times \text{RanInt}\#(2, M - 2) + Fx:$$

$$y = (1 - F)x + Fy:$$

$$C = (1 - F) \times \text{RanInt}\#(1, M - 1) + FC:$$

$$x = x^2 + C - M\text{Int}\left(\frac{x^2 + C}{M}\right):$$

$$y = y^2 + C - M\text{Int}\left(\frac{y^2 + C}{M}\right):$$

$$y = y^2 + C - M\text{Int}\left(\frac{y^2 + C}{M}\right):$$

$$D = \text{GCD}(|x - y|, M):$$

$$F = \text{Int}\left(\frac{1}{D}\right):$$

$$0 \div \left(F + \text{Int}\left(\frac{D}{M}\right)\right)$$

Giá trị ban đầu: $M =$ số n cần phân tích, $F = 0$, các biến còn lại mang giá trị **nguyên không âm** tùy ý.

Cách dùng: Lặp đa câu lệnh tới khi gặp **Math ERROR** thì dừng, kết quả trong biến D .

Lưu ý: Nếu M là số nguyên tố thì thuật toán sẽ chạy vô tận. Đây cũng chính là một điểm yếu lớn của thuật toán!