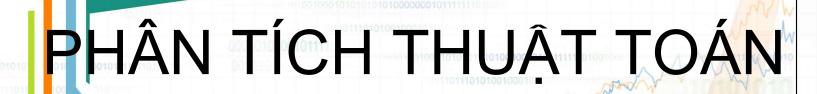
Phân tích & Thiết kế thuật toán (Algorithms Design & Analysis)

L/O/G/O

GV: HUYNH THỊ THANH THƯƠNG

Email: thuonghtt@uit.edu.vn



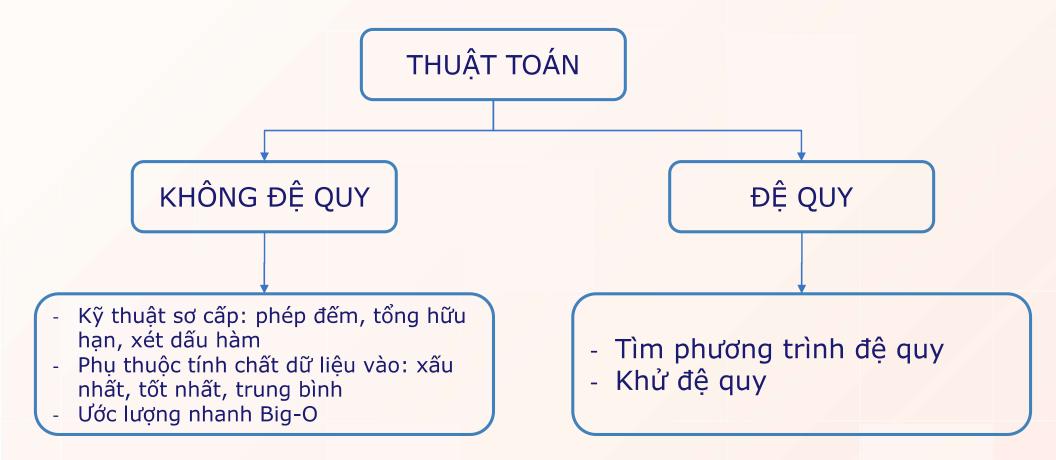
CHƯƠNG 2



L/O/G/O

www.themegallery.com

Đánh giá tính hiệu quả về thời gian



Một số thuật toán thông dụng

Phân tích thuật toán đệ quy (Analysis of Recursive Algorithms)



Cách 1:

- Thành lập phương trình đệ quy
- Giải phương trình đệ quy
 - Thời gian thực hiện chương trình = nghiệm của phương trình đệ quy

Cách 2:

 Khử đệ quy, phân tích Thuật toán không đệ quy

Thành lập phương trình đệ quy (Recurrence relation)



Dạng tổng quát của phương trình đệ qui

$$T(n) = \begin{cases} C(n) \\ g(T(k)) + f(n) \end{cases}$$

- C(n): thời gian thực hiện chương trình trong trường hợp dừng
- g(T(k)) là đa thức của T(k)
- f(n) là thời gian phân chia/kết hợp các kết quả

Thành lập phương trình đệ quy (Recurrence relation)

Vi trùng cứ 1 giờ lại nhân đôi. Vậy sau 5 giờ sẽ có mấy con vi trùng nếu ban đầu có 2 con?

$$T(n) = \begin{cases} C_1 & khi \ n = 0 \\ T(n-1) + C_2 & khi \ n > 0 \end{cases}$$

```
int vitrung(int n)
{
    if (n == 0)
       return 2;
    return 2*vitrung(n-1);
}
```





- ❖ Dùng đệ quy để thay thế T(k) (k < n) vào phía phải của phương trình
- Łặp lại cho đến khi tất cả T(k) được thay thế bởi các biểu thức T(1) hoặc T(0).
- ❖ T(1) và T(0) luôn là hằng số
- → công thức T(n) chứa các số hạng chỉ liên quan tới n và hằng số.

Giải phương trình đệ quy (Solve the recurrence relation)



Giải bằng Thay thế:

$$T(n) = \begin{cases} C_1 & khi \ n = 1 \\ 2T(\frac{n}{2}) + nC_2 & khi \ n > 1 \end{cases}$$

$$T(n) = 2T\left(\frac{n}{2}\right) + nC_2$$

$$= 2\left[2T\left(\frac{n}{4}\right) + \frac{n}{2}C_2\right] + nC_2 = 4T\left(\frac{n}{4}\right) + 2nC_2$$

$$= 4\left[2T\left(\frac{n}{8}\right) + \frac{n}{4}C_2\right] + 2nC_2 = 8T\left(\frac{n}{8}\right) + 3nC_2$$
....
$$= 2^i T\left(\frac{n}{2^i}\right) + inC_2$$





• Quá trình kết thúc khi $\frac{n}{2^i} = 1 \rightarrow i = \log n$

*Khi đó:
$$T(n) = 2^{i}T\left(\frac{n}{2^{i}}\right) + inC_{2}$$
$$= 2^{\log n}T\left(\frac{n}{2^{\log n}}\right) + (\log n)nC_{2}$$
$$= nT(1) + n(\log n)C_{2}$$

$$T(n) = nC_1 + n(\log n)C_2$$

Phương pháp Truy hồi/Thay thể (Backward Substitution)

Bài tập trên lớp: Lấy điểm quá trình (Đã sửa tại lớp)

$$T(n) = \begin{cases} c_1 & khi \ n = 0 \\ T(n-1) + n + c_2 & khi \ n > 0 \end{cases}$$

Phân tích thuật toán đệ quy - Giải phương trình đệ quy



- Truy hồi/Thay thế
- Phương trình đặc trưng
- Phương pháp hàm sinh
- Dịnh lý Master
- Doán nghiệm và quy nạp
- Khác

- Backward substitution
- Characteristic equation
- Generating function
- Master theorem
- Guessing and Induction

Xét phương trình dạng:

$$a_0 T(n) + a_1 T(n-1) + ... + a_k T(n-k) = 0$$
 (1)

⊕Đặt T(n) = Xⁿ, đưa (1) về dạng phương trình ẩn X:

$$a_0 X^n + a_1 X^{n-1} + ... + a_k X^{n-k} = 0$$

$$X^{n-k} (a_0 X^k + a_1 X^{k-1} + ... + a_k) = 0$$

$$a_0 X^k + a_1 X^{k-1} + \dots + a_k = 0$$
 (2)

Xét phương trình dạng:

$$a_0 T(n) + a_1 T(n-1) + ... + a_k T(n-k) = 0$$
 (1)

◆ Đặt T(n) = Xⁿ, đưa (1) về dạng phương trình ẩn X:

Tìm nghiệm của pt đặc trưng:

$$a_0 X^k + a_1 X^{k-1} + ... + a_k = 0$$
 (2)

(2) là phương trình đặc trưng bậc k của phương trình truy hồi (1)

- Tìm nghiệm của phương trình đặc trưng
- > TH1: (2) có nghiệm đơn
 - Giả sử X₁, X₂ là các nghiệm đơn của (2) thì
 - $T(n) = c_1 X_1^n + c_2 X_2^n + \dots \text{ với } c_i \text{ là các hằng}$
- > TH2: (2) có nghiệm bội
 - Giả sử u là nghiệm bội m của (2) thì
 - $T(n) = c_1 u^n + c_2 n u^n + c_3 n^2 u^n ... + c_m n^{m-1} u^n + ...$

❖ Ví dụ:

•
$$T(n) = 5T(n-1) - 8T(n-2) + 4T(n-3) va$$

•
$$T(0) = 0$$
, $T(1) = 1$, $T(2) = 2$

Giải:

Bước 1: Tìm phương trình đặc trưng

Xét phương trình:
$$T(n) - 5T(n-1) + 8T(n-2) - 4T(n-3) = 0$$

Đặt
$$X^n = T(n)$$

Ta có:
$$X^n - 5X^{n-1} + 8X^{n-2} - 4X^{n-3} = 0$$

Phương trình đặc trưng :
$$X^3 - 5X^2 + 8X - 4 = 0$$
 (*)

$$(X-1)(X-2)^2 = 0$$

Bước 2: Giải phương trình đặc trưng:

Phương trình đặc trưng : $X^3 - 5X^2 + 8X - 4 = 0 \Leftrightarrow (X-1)(X-2)^2 = 0$

có 1 nghiệm đơn $X_1 = 1$ và nghiệm kép $X_2 = 2$

$$T(n) = c_1 X_1^n + c_2 X_2^n + c_3 n X_2^n$$

$$T(n) = c_1 1^n + c_2 2^n + c_3 n 2^n$$

--> dựa vào T(0), T(1), T(2) để tìm các tham số

$$T(n) = c_1 1^n + c_2 2^n + c_3 n 2^n$$

Bước 3: Tìm các tham số

Ta có

•
$$T(0) = 0$$
 $\rightarrow c_1 + c_2 = 0$

•
$$T(1) = 1$$
 $\rightarrow c_1 + 2c_2 + 2c_3 = 1$

•
$$T(2) = 2$$
 $\rightarrow c_1 + 4c_2 + 8c_3 = 2$

Giải hệ phương trình: $c_1 = -2$, $c_2 = 2$, $c_3 = -1/2$

Kết luận:
$$T(n) = 2^{n+1} - n2^{n-1} - 2$$

Phân tích thuật toán đệ quy - Giải phương trình đệ quy



- Truy hồi/Thay thế
- Phương trình đặc trưng
- Phương pháp hàm sinh
- Dịnh lý Master
- Đoán nghiệm và quy nạp
- Khác

- Backward substitution
- Characteristic equation
- Generating function
- Master theorem
- Guessing and Induction