



$O(f(x))$ và đánh giá thời gian thực hiện thuật toán.

Bởi:

Khoa CNTT ĐHSPTK Hưng Yên

Khi đánh giá thời gian thực hiện bằng phương pháp toán học, chúng ta sẽ bỏ qua nhân tố phụ thuộc vào cách cài đặt chỉ tập trung vào xác định độ lớn của thời gian thực hiện $T(n)$.

Giả sử n là số nguyên không âm. $T(n)$ và $f(n)$ là các hàm thực không âm. Ta viết $T(n) = O(f(n))$ (đọc $T(n)$ là ô lớn của $f(n)$), nếu và chỉ nếu tồn tại các hằng số dương c và n_0 sao cho $T(n) \leq c \cdot f(n)$, với mọi $n \geq n_0$.

Nếu một thuật toán có thời gian thực hiện $T(n) = O(f(n))$, ta nói thuật toán có thời gian thực hiện cấp $f(n)$. Từ định nghĩa ký hiệu ô lớn, ta có thể xem rằng hàm $f(n)$ là cận trên của $T(n)$.

Ví dụ. Giả sử $T(n) = 3n^2 + 4n + 5$. Ta có

$$3n^2 + 4n + 5 \leq 3n^2 + 4n^2 + 5n^2 = 12n^2, \text{ với mọi } n \geq 1.$$

Vậy $T(n) = O(n^2)$. Trong trường hợp này ta nói thuật toán có thời gian thực hiện cấp n^2 , hoặc gọn hơn, thuật toán có thời gian thực hiện bình phương.

Để dàng thấy được, nếu $T(n) = O(f(n))$ và $f(n) = O(f_1(n))$, thì $T(n) = O(f_1(n))$. Thật vậy, vì $T(n)$ là ô lớn của $f(n)$ và $f(n)$ là ô lớn của $f_1(n)$ nên tồn tại các hằng số c_0, n_0, c_1, n_1 sao cho $T(n) \leq c_0 f(n)$ với mọi $n \geq n_0$ và $f(n) \leq c_1 f_1(n)$ với mọi $n \geq n_1$. Từ đó ta có $T(n) \leq c_0 c_1 f_1(n)$ với mọi $n \geq \max(n_0, n_1)$.

Khi biểu diễn cấp của thời gian thực hiện thuật toán bởi hàm $f(n)$, chúng ta sẽ chọn $f(n)$ là hàm nhỏ nhất, đơn giản nhất có thể được sao cho $T(n) = O(f(n))$. Thông thường $f(n)$ là các hàm số sau đây: $f(n) = 1$; $f(n) = \log n$; $f(n) = n$; $f(n) = n \log(n)$; $f(n) = n^2$; $n^3 \dots$; $f(n) = 2^n$.

$O(f(x))$ và đánh giá thời gian thực hiện thuật toán.

- Nếu $T(n) = O(1)$ điều này có nghĩa là thời gian thực hiện thuật toán được chặn trên bởi một hằng nào đó, trong trường hợp này ta nói thuật toán có thời gian *thực hiện hằng*.

- Nếu $T(n) = O(n)$, tức là bắt đầu từ một n_0 nào đó trở đi ta có $T(n) \leq cn$ với một hằng số c nào đó, trong trường hợp này ta nói thuật toán có thời gian *thực hiện tuyến tính*.

Bảng sau đây cho ta các cấp thời gian thực hiện thuật toán được sử dụng rộng rãi nhất và tên gọi của chúng.

Ký hiệu $O(f(x))$ của $f(x)$	Độ phức tạp loại
$O(1)$	Hằng
$O(\log)$	Logarit
$O(n)$	Tuyến tính
$O(n \log n)$	$n \log n$
$O(n^2)$	Bình phương
$O(n^3)$	Lập phương
$O(2^n)$	Mũ
$O(n!)$	Giai thừa

Danh sách trên sắp xếp theo thứ tự tăng dần của hàm thời gian thực hiện.

- Các hàm loại 2^n , $n!$, nn thường được gọi là các hàm loại mũ. Thuật toán với thời gian chạy có cấp hàm loại mũ thì tốc độ rất chậm

- Các hàm n , n^3 , n^2 , $n \log 2 n$ thường được gọi là các hàm đa thức. Thuật toán với thời gian chạy có cấp hàm đa thức thường chấp nhận được