

Chương 1

Tổng quan về phân tích giải thuật

PART 2

Nội dung

Một số ví dụ về phân tích độ phức tạp thuật toán

Ôn lại

$$O(f) = \{g \mid \exists N, c \in \mathbb{N}. \forall n > N. g(n) \leq cf(n)\}$$

$$o(f) = \{g \mid \forall c \in \mathbb{N} \exists N \in \mathbb{N}. \forall n > N. cg(n) \leq cf(n)\}$$

$$\Theta(f) = \{g \mid g \in O(f) \wedge g \notin o(f)\}$$

$$\Omega(f) = \{g \mid f \in O(g)\}$$

$$\omega(f) = \{g \mid f \in o(g)\}$$

Ôn lại

Các hàm số đo độ phức tạp cơ bản

Phụ thuộc vào số lượng các dòng lệnh được thực hiện sẽ ảnh hưởng đến thời gian chạy của thuật toán hay chương trình

Tuyến tính: $O(n)$

Logarit: $O(\log n)$

Bình phương: $O(n^2)$

Hàm mũ: $O(2^n)$

Ôn lại

So sánh các hàm số thời gian

Logarit: $O(\log n)$ < Tuyến tính: $O(n)$

< Bình phương: $O(n^2)$ < Hàm mũ: $O(2^n)$

N = 1000	Nanosecond speed
Logarith	10 Nanoseconds
Tuyến tính	1 microseconds
Bình Phương	1 miliseconds
Hàm Mũ	10^{204} years

Ví dụ 1:

```
public int ex1 (int a, int b)
{
    int ans = 1;
    while (b>0) {
        ans *=a;
        b -=1;
    }
    return ans;
}
//time = 2 + 3b
```

Ví dụ 2:

```
public int ex2(int a, int b)
{
    if(b==1) {
        return a;
    } else {
        return a * ex2(a,b-1);
    }
}
//Time = 3b - 1
```

$t(b) = 3 + t(b - 1)$
 $\Rightarrow t(b) = 3 + t(b-1)$
 $\Rightarrow t(b) = 3 + 3 + t(b-2)$
 $\Rightarrow t(b) = 3 + 3 + 3 + t(b-3)$
 $\Rightarrow \dots$
 $\Rightarrow t(b) = 3*k + t(b-k)$
 $\Rightarrow \text{end: } b-k=1 \Rightarrow k=b-1$
 $\Rightarrow 3*b - 1$
 $\Rightarrow O(b)$

THANK YOU