INF 02779 Análise de Algoritmos - 2002/2	11/27/02
Aula 4: Notação Assintótica	
Instrutor: Berilhes Borges Garcia	Escriba: Alexandre Vervloet

DRAFT

1 Crescimento de Funções

- *Notação Assintótica.*

  Dada uma entrada grande o bastante, examine somente a ordem de crescimento do tempo de execução. Isto é, como o tempo de execução aumenta com o tamanho da entrada no limite.

  Usualmente definida somente para tamanho de entradas inteiras.

  Também consideraremos alguns excessos de notação que serão bastante utilizados.

2 Notação Assintótica

- Definida em termos de funções cujos domínios são o conjunto dos números naturais.
- Utilizada para descrever a função tempo de execução no pior caso $T(n)$, usualmente definida somente para tamanhos de entradas inteiras.

3 Notação Θ

Definição 1. $\Theta(g(n))$={*f(n): existem constantes positivas* c_1, c_2 *e* n_0 *tal que* $0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n)$*, para todo* $n \geq n_0$ }.

A definição anterior afirma que uma função $f(n)$ pertence ao conjunto $\Theta(g(n))$ se existem constantes inteiras positivas c_1 e c_2 de modo que essa possa ser inserida entre $c_1 g(n)$ e $c_2 g(n)$, para valores de n suficientemente grandes. Isto requer que $\Theta(g(n))$ seja assintóticamente não negativa.

Nós dizemos que $g(n)$ é um limite assintótico justo para $f(n)$:

$$0 < lim_{n \to \infty}[f(n)|g(n)] < \infty \tag{1}$$

4 Notação O

Definição 2. *Para uma dada função* $g(n)$*, nós denotamos por* $O(g(n))$ *o conjunto de funções:* $O(g(n))$={*f(n): existem constantes positivas c e* n_0 *tal que* $0 \leq f(n) \leq cg(n)$*, para todo* $n \geq n_0$ }.

Nós podemos dizer que $g(n)$ é um limite superior assintótico para $f(n)$:

$$0 < lim_{n \to \infty}[f(n)|g(n)] < \infty \tag{2}$$

- Em termos de conjuntos, $\Theta(g(n)) \subseteq O(g(n))$

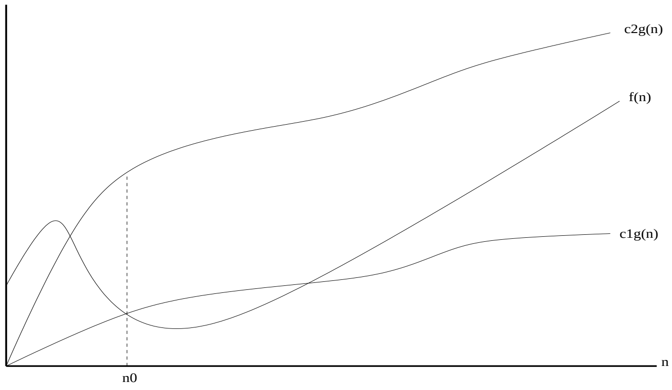

Figure 1: Exemplo de $f(n) = \Theta(g(n))$

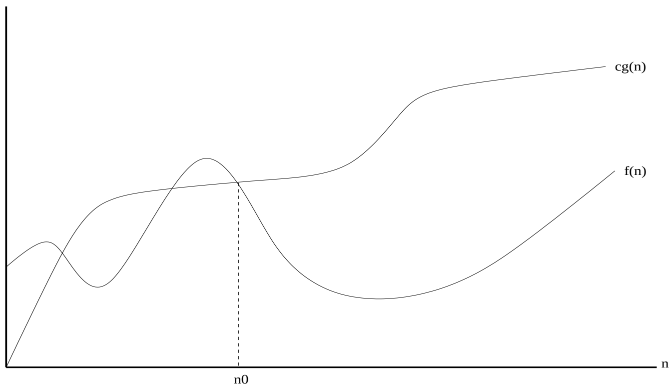

Figure 2: Exemplo de $f(n) = \emptyset(g(n))$

5 Notação Ω

Definição 3. *Para uma dada função $g(n)$, nós denotamos por $\Omega(g(n))$ o conjunto de funções.*

$\Omega(g(n))$=*{f(n): existem constantes positivas c e n_0 tal que $0 \leq cg(n) \leq f(n)$, para todo $n \geq n_0$ }.*

Nós dizemos que $g(n)$ é um limite assintótico justo para $f(n)$:

$$0 < lim_{n \to \infty}[f(n)|g(n)] \tag{3}$$

Observação 1. *Para qualquer das duas funções $g(n)$, $f(n)$, $f(n) = \Theta(g(n))$ se e somente se $f(n) = O(g(n))$ e $f(n) = \Omega(g(n))$.*

6 Notação Assintótica em Equações

- Utilizada para substituir funções de termo de ordem mais baixa de modo a simplificar equações expressões.
- Por exemplo, $4n^3 + 3n^2 + 2n + 1 = 4n^3 + 3n^2 + \Theta(n) = 4n^3 + \Theta(n^2) = \Theta(n^3)$

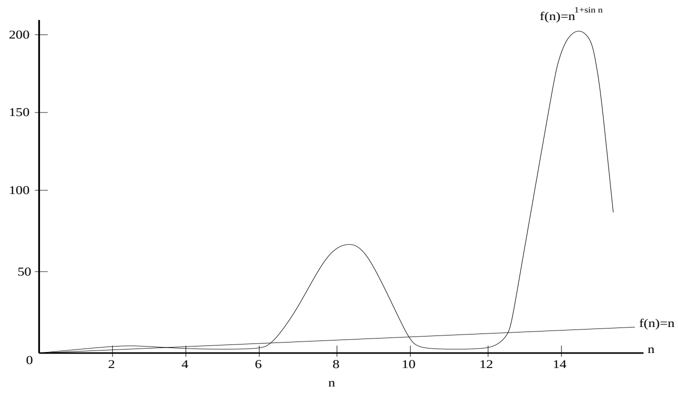

Figure 3: Exemplo de $f(n) \neq \emptyset(g(n))$ e $f(n) \neq \Omega(g(n))$

- Ou nós podemos fazer o seguinte: $4n^3 + 3n^2 + 2n + 1 = 4n^3 + f(n^2)$ onde $f(n^2)$ simplifica a equacao

7 Notação-o

Definição 4. *Para uma dada função $g(n)$, nós denotamos por $o(g(n))$ o conjunto de funções: $o(g(n))$={f(n): para qualquer constante positiva $c > 0$, existe uma constante n_0 tal que $0 \leq f(n) < cg(n)$, para todo $n \geq n_0$ }.*

$f(n)$ torna-se insignificante com relação à $g(n)$ quando n tende para infinito:

$$lim_{n \to \infty}[f(n)|g(n)] = 0 \tag{4}$$

Nós podemos dizer que $g(n)$ é um limite superior para $f(n)$ que não é assintoticamente justo.

8 Notação-ω

Definição 5. *Para uma dada função $g(n)$, nós denotamos por $\omega(g(n))$ o conjunto de funções: $\omega(g(n))$={f(n): para qualquer constante positiva $c > 0$, existe uma constante n_0 tal que $0 \leq cg(n) < f(n)$, para todo $n \geq n_0$ }.*

$f(n)$ torna-se arbitrariamente grande com relação à $g(n)$ quando n tende para infinito:

$$lim_{n \to \infty}[f(n)|g(n)] = \infty \tag{5}$$

Nós dizemos que $g(n)$ é um limite inferior para $f(n)$ que não é assintoticamente justo.

9 Propriedades

- *Transitividade*

  $f(n) = \Theta(g(n))$ & $g(n) = \Theta(h(n)) \rightarrow f(n) = \Theta(h(n))$

  $f(n) = O(g(n))$ & $g(n) = O(h(n)) \rightarrow f(n) = O(h(n))$

  $f(n) = \Omega(g(n))$ & $g(n) = \Omega(h(n)) \rightarrow f(n) = \Omega(h(n))$

  $f(n) = o(g(n))$ & $g(n) = o(h(n)) \rightarrow f(n) = o(h(n))$

  $f(n) = \omega(g(n))$ & $g(n) = \omega(h(n)) \rightarrow f(n) = \omega(h(n))$

- *Simetria*

  $f(n) = \Theta(g(n))$ se e somente se $g(n) = \Theta(f(n))$

- *Simetria transposta*

  $f(n) = O(g(n))$ se e somente se $g(n) = \Omega(f(n))$

  $f(n) = o(g(n))$ se e somente se $g(n) = \omega(f(n))$

10 Comparações de Funções

$$
\begin{gathered}
f(n) = O(g(n)) \sim a \leq b \\
f(n) = \Omega(g(n)) \sim a \geq b \\
f(n) = \Theta(g(n)) \sim a = b \\
f(n) = o(g(n)) \sim a < b \\
f(n) = \omega(g(n)) \sim a > b
\end{gathered}
$$

11 Duas Funções que não podem ser comparadas

Aqui entra a fig04.ps

12 Exemplos

A B

- $5n^2 + 100n$ $3n^2 + 2$ $A \in \Theta(B)$
- $log_3(n^2)$ $log_2(n^3)$ $A \in \Theta(B)$
- $n^{lg(4)}$ $3^{lg(n)}$ $A \in \omega(B)$
- $lg^2(n)$ $n^{\frac{1}{2}}$ $A \in o(B)$

13 Exemplos

A B

- $5n^2 + 100n$ $3n^2 + 2$ $A \in \Theta(B)$

  $A \in \Theta(n^2)$, $n^2 \in \Theta(B) \rightarrow A \in \Theta(B)$

- $log_3(n^2)$ $log_2(n^3)$ $A \in \Theta(B)$

  $log_b a = \frac{log_c a}{log_c b}$; $A = \frac{2lg(n)}{lg(3)}$, $B = 3lg(n)$, $\frac{A}{B} = \frac{2}{3lg(3)}$

- $n^{lg(4)}$ $3^{lg(n)}$ $A \in \omega(B)$

  $a^{logb} = b^{loga}$; $B = 3^{lg(n)} = n^{lg(3)}$; $\frac{A}{B} = n^{lg(\frac{4}{3})} \rightarrow \infty$ quando $n \rightarrow \infty$

- $lg^2(n)$ $n^{\frac{1}{2}}$ $A \in o(B)$

  $lim_{x \rightarrow \infty}(log^a n | n^b) = 0$, aqui $a = 2$ e $b = \frac{1}{2} \rightarrow A \in o(B)$

14 Revisão sobre Somatórias

- Porque nós necessitamos saber isto?

  Nós necessitamos disto para computar o tempo de execução de um determinado algoritmo.

- Exemplo: Insertion Sort

- Séries constantes:

  $\sum_{i=a}^{b} 1 = b - a + 1$

- Série Quadrática: Para $n \geq 0$,

  $\sum_{i=1}^{n} i^2 = 1^2 + 2^2 + ... + n^2 = (2n^3 + 3n^2 + n)/6$

- Série Geométrica: Para $n \geq 0$,

  $\sum_{i=1}^{n} c^i = c + c^2 + ... + c^n = [c^{(n+1)} - 1]/(c-1)^2$

- Série Geométrica-linear: Para $n \geq 0$,

  $\sum_{i=1}^{n} ic^i = c + 2c^2 + ... + nc^n = [(n-1)c^{(n+1)} - nc^n + c]/(c-1)^2$