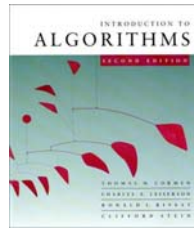




CS 3343 -- Fall 2007



The Master Theorem

Carola Wenk

Slides courtesy of Charles Leiserson with small changes by Carola Wenk

9/18/07

CS 3343 Analysis of Algorithms

1



The master method

The master method applies to recurrences of the form

$$T(n) = aT(n/b) + f(n),$$

where $a \geq 1$, $b > 1$, and f is asymptotically positive.

9/18/07

CS 3343 Analysis of Algorithms

2



Master theorem

$$T(n) = aT(n/b) + f(n)$$

CASE 1: $f(n) = O(n^{\log_b a - \epsilon})$

$$\Rightarrow T(n) = \Theta(n^{\log_b a}).$$

CASE 2: $f(n) = \Theta(n^{\log_b a})$

$$\Rightarrow T(n) = \Theta(n^{\log_b a} \log n).$$

CASE 3: $f(n) = \Omega(n^{\log_b a + \epsilon})$ and $af(n/b) \leq cf(n)$

$$\Rightarrow T(n) = \Theta(f(n)).$$

Merge sort: $a = 2, b = 2 \Rightarrow n^{\log_b a} = n$

$$\Rightarrow \text{CASE 2 } (k = 0) \Rightarrow T(n) = \Theta(n \log n).$$

9/18/07

CS 3343 Analysis of Algorithms

3



Three common cases

Compare $f(n)$ with $n^{\log_b a}$:

1. $f(n) = O(n^{\log_b a - \epsilon})$ for some constant $\epsilon > 0$.

- $f(n)$ grows polynomially slower than $n^{\log_b a}$ (by an n^ϵ factor).

Solution: $T(n) = \Theta(n^{\log_b a}).$

2. $f(n) = \Theta(n^{\log_b a}).$

- $f(n)$ and $n^{\log_b a}$ grow at similar rates.

Solution: $T(n) = \Theta(n^{\log_b a} \log n).$

9/18/07

CS 3343 Analysis of Algorithms

4



Three common cases (cont.)

Compare $f(n)$ with $n^{\log_b a}$:

3. $f(n) = \Omega(n^{\log_b a + \varepsilon})$ for some constant $\varepsilon > 0$.
- $f(n)$ grows polynomially faster than $n^{\log_b a}$ (by an n^ε factor),
- and $f(n)$ satisfies the **regularity condition** that $af(n/b) \leq cf(n)$ for some constant $c < 1$.

Solution: $T(n) = \Theta(f(n))$.



Examples

Ex. $T(n) = 4T(n/2) + n$
 $a = 4, b = 2 \Rightarrow n^{\log_b a} = n^2; f(n) = n$.
CASE 1: $f(n) = O(n^{2-\varepsilon})$ for $\varepsilon = 1$.
 $\therefore T(n) = \Theta(n^2)$.

Ex. $T(n) = 4T(n/2) + n^2$
 $a = 4, b = 2 \Rightarrow n^{\log_b a} = n^2; f(n) = n^2$.
CASE 2: $f(n) = \Theta(n^2)$.
 $\therefore T(n) = \Theta(n^2 \log n)$.



Examples

Ex. $T(n) = 4T(n/2) + n^3$
 $a = 4, b = 2 \Rightarrow n^{\log_b a} = n^2; f(n) = n^3$.
CASE 3: $f(n) = \Omega(n^{2+\varepsilon})$ for $\varepsilon = 1$
and $4(n/2)^3 \leq cn^3$ (reg. cond.) for $c = 1/2$.
 $\therefore T(n) = \Theta(n^3)$.

Ex. $T(n) = 4T(n/2) + n^2/\log n$
 $a = 4, b = 2 \Rightarrow n^{\log_b a} = n^2; f(n) = n^2/\log n$.
Master method does not apply. In particular,
for every constant $\varepsilon > 0$, we have $n^\varepsilon = \omega(\log n)$.



Extended Case 2

Compare $f(n)$ with $n^{\log_b a}$:

$f(n) = \Theta(n^{\log_b a} \log^k n)$ for some constant $k \geq 0$.

- $f(n)$ and $n^{\log_b a}$ grow at similar rates.

Solution: $T(n) = \Theta(n^{\log_b a} \log^{k+1} n)$.