Master Theorem

- Motivation
- Master Theorem
- Examples
- Group activities
Motivation

● We only care about asymptotic behavior when analyzing algorithms.
● Rather than analyzing exactly the recurrence relation, we only need to find asymptotic behavior.
● Master Theorem is a great tool to do that!
Master theorem

The master method applies to recurrences of the form

\[ T(n) = a \ T(n/b) + f(n) , \]

where \( a > 1, b > 1, \) and \( f \) is asymptotically positive.
Master theorem

\[ T(n) = \begin{cases} 
\Theta(n^{\log_b a}) & f(n) = O(n^{\log_b a - \varepsilon}) \\
\Theta(n^{\log_b a \log n}) & f(n) = \Theta(n^{\log_b a}) \\
\Theta(f(n)) & f(n) = \Omega(n^{\log_b a + \varepsilon}) \text{ AND } af(n/b) < cf(n) \text{ for large } n \\
\end{cases} \]

\[ \varepsilon > 0 \]

\[ c < 1 \]
Example 1

\[ T(n) = 4T(n/2) + n \]
Example 1

\[ T(n) = 4T(n/2) + n \]

\[ a = 4, \ b = 2 \Rightarrow n^{\log_b(a)} = n^2; \quad f(n) = n. \]

CASE 1: \( f(n) = O(n^{2-\varepsilon}) \) for \( \varepsilon = 1 \).

\[ T(n) = (n^2). \]
Example 2

$$T(n) = 4T(n/2) + n^2$$
Example 2

\[ T(n) = 4T(n/2) + n^2 \]

\[ a = 4, \ b = 2 \Rightarrow n^{\log_b(a)} = n^2; \ f(n) = n^2. \]

CASE 2: \( f(n) = \Theta(n^2). \)

\[ \rightarrow T(n) = (n^2 \log n). \]
Example 3

\[ T(n) = 4T(n/2) + n^3 \]
Example 3

$T(n) = 4T(n/2) + n^3$

$a = 4, b = 2 \Rightarrow n^{\log_b(a)} = n^2; f(n) = n^3$.

CASE 3: $f(n) = \Omega(n^2 + \varepsilon)$ for $\varepsilon = 1$.

and $4(n/2)^3 \leq cn^3$ for $c = 1/2$.

$\rightarrow T(n) = (n^3)$. 
Example 4

\[ T(n) = 4T(n/2) + n^2 / \log n \]
Example 4

\[ T(n) = 4T(n/2) + n^2 / \log n \]

\[ a = 4, \ b = 2 \Rightarrow n^{\log_b(a)} = n^2; \ f(n) = n^2 / \log n. \]

Master theorem does not apply. In particular, for every constant \( \varepsilon > 0 \), we have \( n^\varepsilon = \omega(\log n) \).
Group activities

Solve and discuss several recurrence relations.