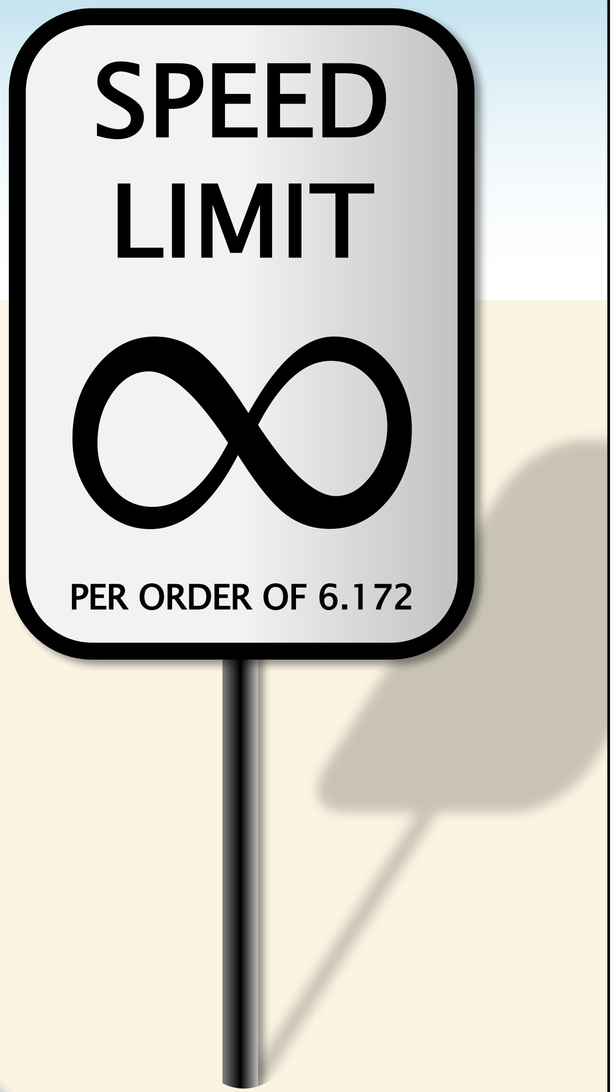
6.172
Performance
Engineering
of Software
Systems



REVIEW

Divide-and-Conquer Recurrences

Charles E. Leiserson

October 4, 2021

The Master Method

The **master method** for solving divide-and-conquer recurrences applies to recurrences of the form*

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

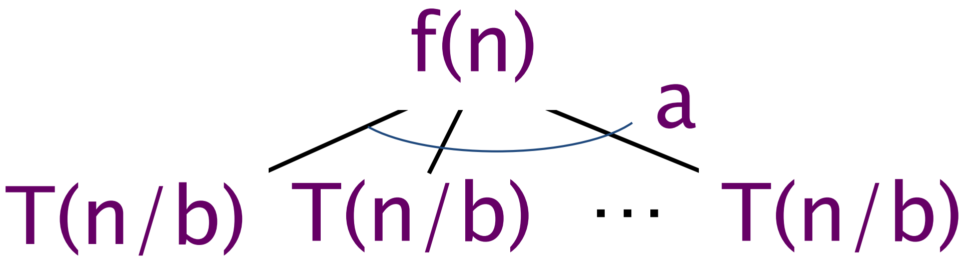
where $a \ge 1$, $b > 1$, and f is asymptotically positive.

*The unstated base case is $T(n) = \Theta(1)$ for all $n \le n_0$ for a sufficiently large constant n_0.

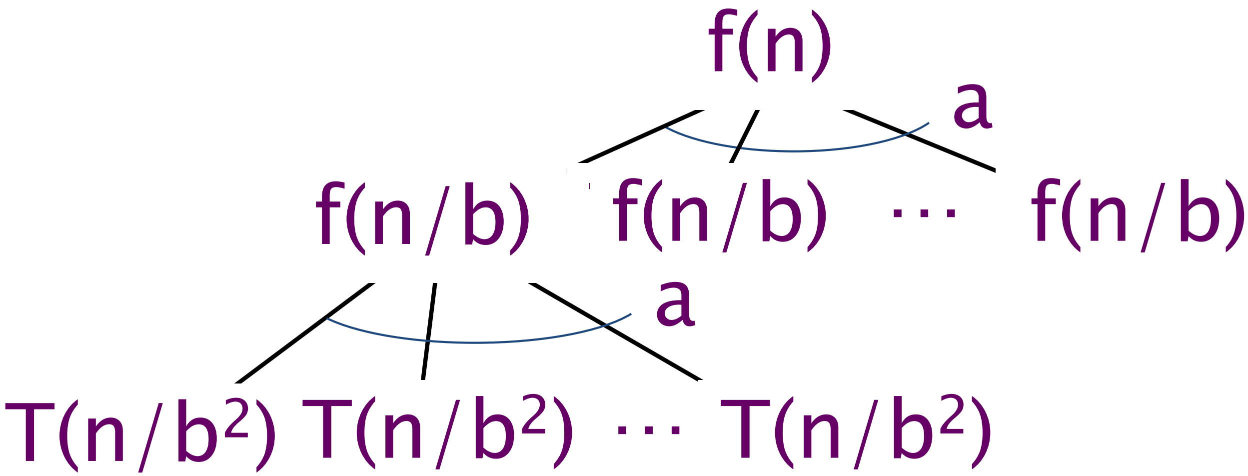
Recursion Tree: $T(n) = aT(n/b) + f(n)$

$T(n)$

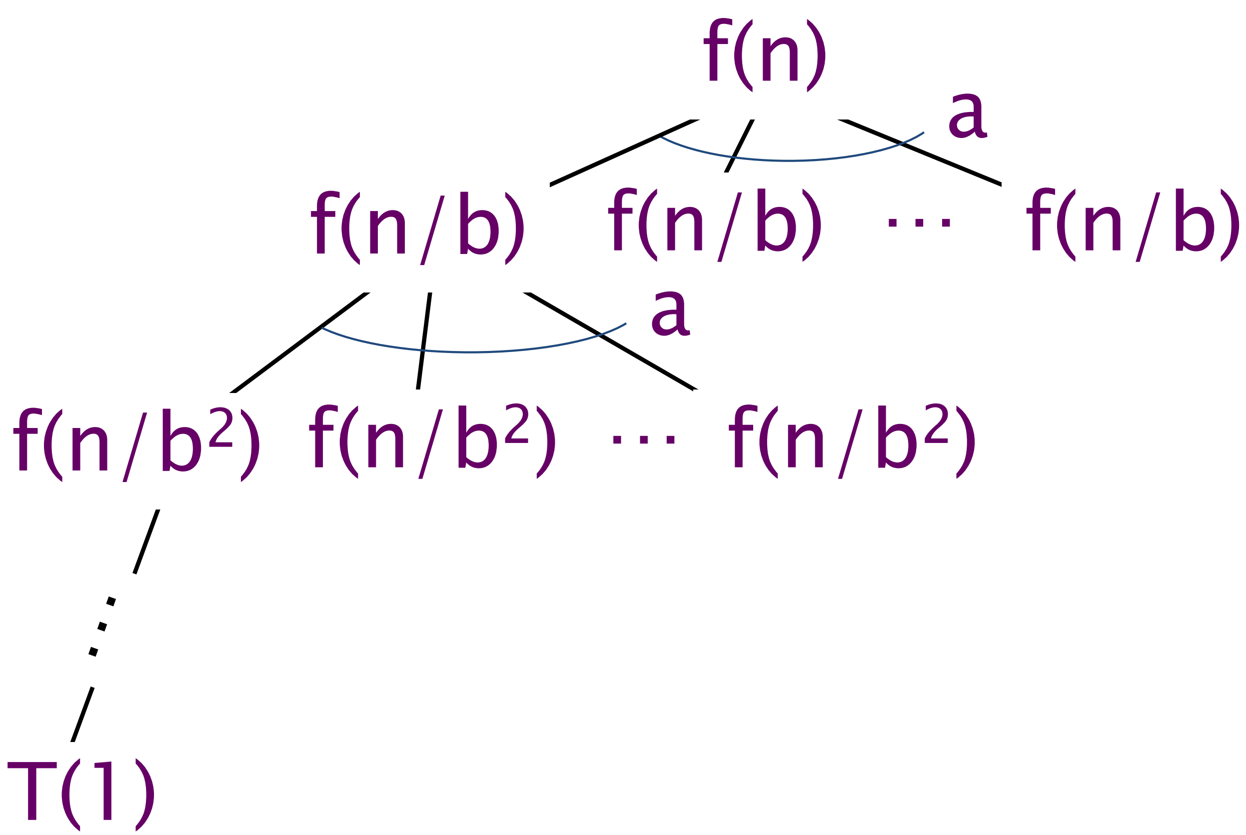
Recursion Tree: $T(n) = aT(n/b) + f(n)$



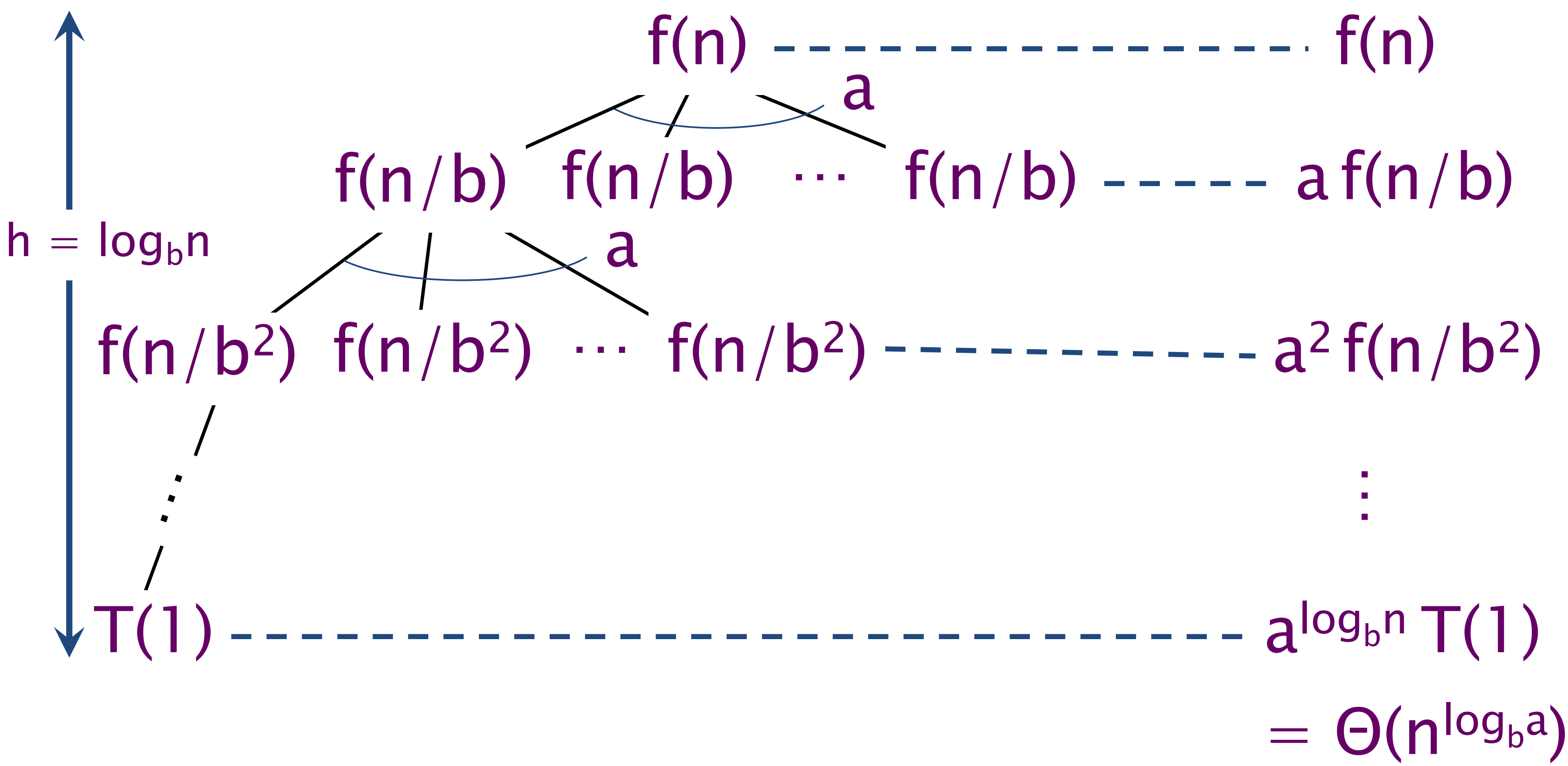
Recursion Tree: $T(n) = aT(n/b) + f(n)$



Recursion Tree: $T(n) = aT(n/b) + f(n)$

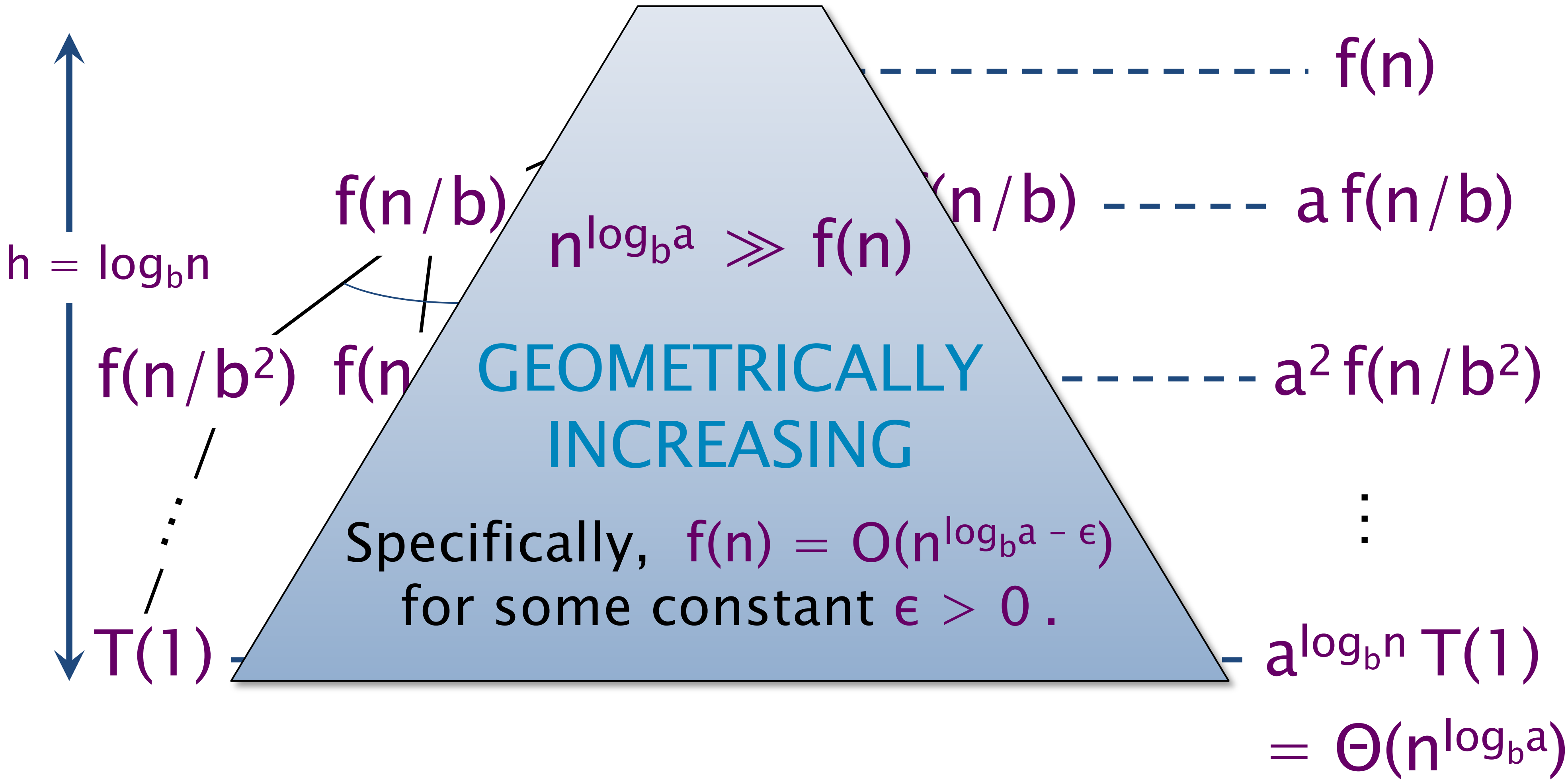


Recursion Tree: $T(n) = aT(n/b) + f(n)$



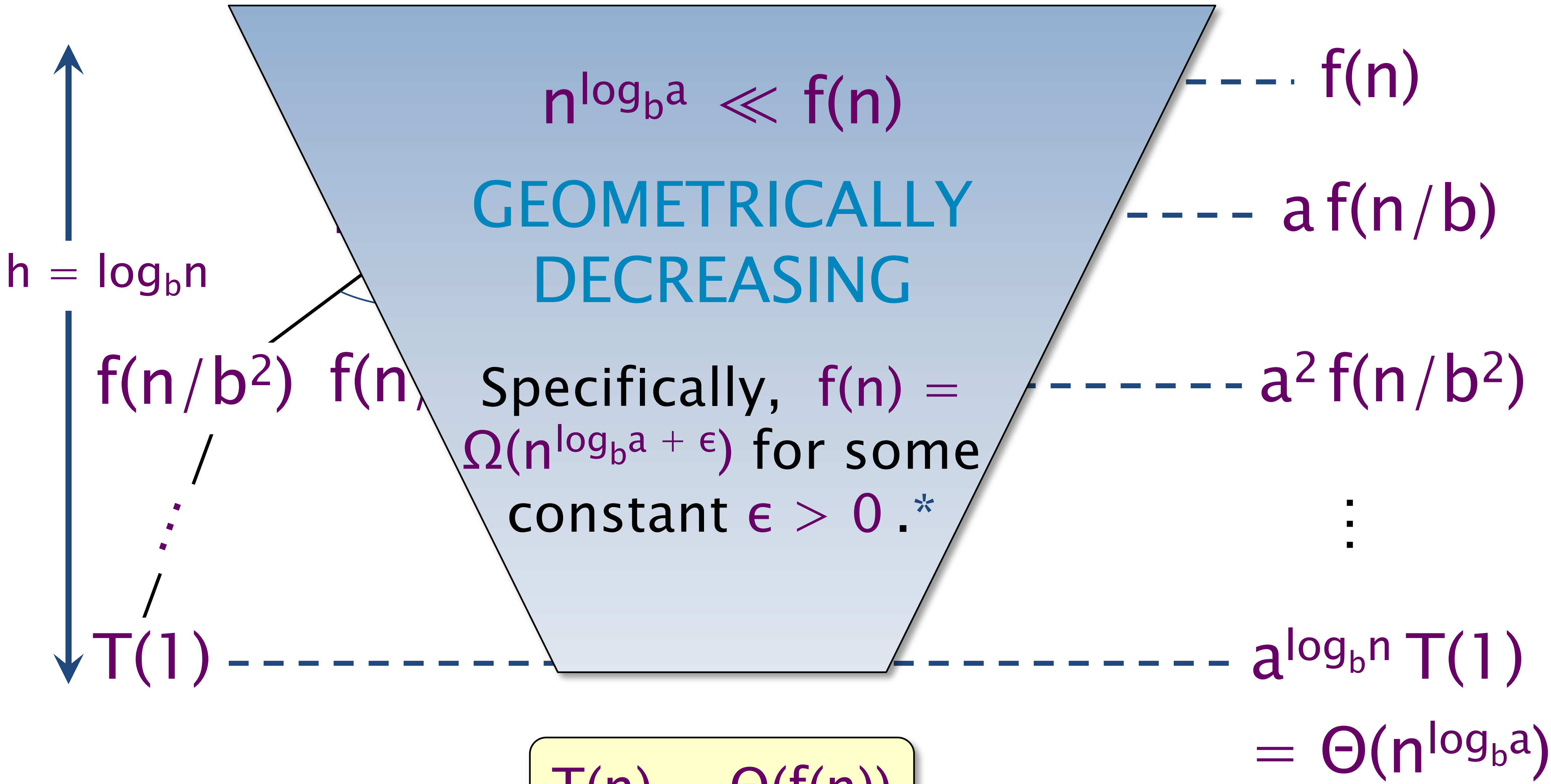
IDEA: Compare $n^{\log_b a}$ with $f(n)$.

Master Method — Case I



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

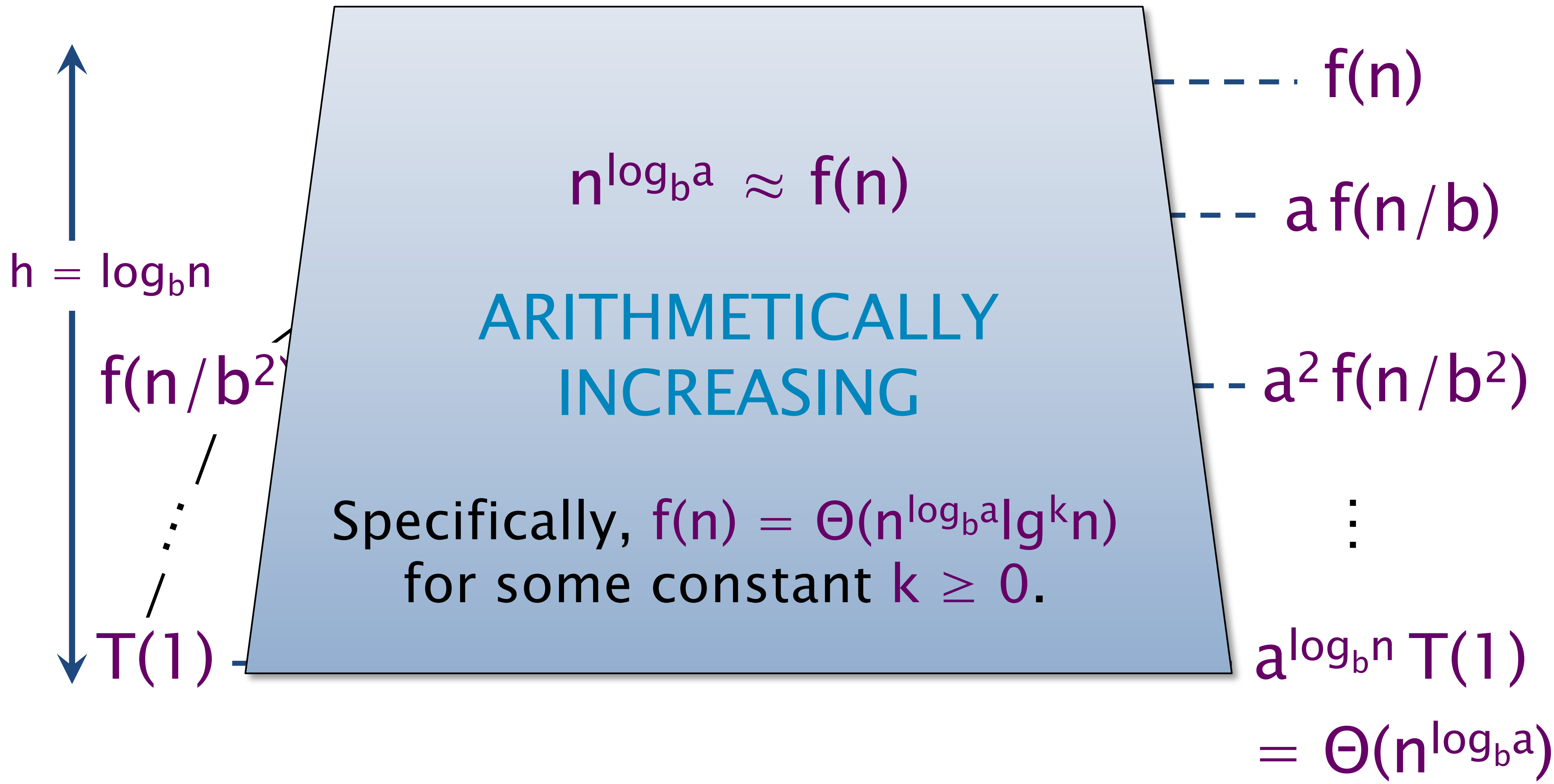
Master Method — CASE 3



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

*and $f(n)$ satisfies the regularity condition that $a\,f(n/b) \le c\,f(n)$ for some constant $c < 1$.

Master Method — CASE 2



$$T(n) = \Theta(n^{\log_b a} \lg^{k+1} n)$$

Master–Method Cheat Sheet

Solve

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

where $a \geq 1$ and $b > 1$.

CASE 1: $f(n) = O(n^{\log_b a - \epsilon})$, constant $\epsilon > 0$
$\Rightarrow T(n) = \Theta(n^{\log_b a})$.

CASE 2: $f(n) = \Theta(n^{\log_b a} \lg^k n)$, constant $k \geq 0$
$\Rightarrow T(n) = \Theta(n^{\log_b a} \lg^{k+1} n)$.

CASE 3: $f(n) = \Omega(n^{\log_b a + \epsilon})$, constant $\epsilon > 0$
(and regularity condition)
$\Rightarrow T(n) = \Theta(f(n))$.

https://tinyurl.com/mm-cheat

Master Method Quiz

- $T(n) = 4T(n/2) + n$
  $n^{\log_b a} = n^2 \gg n \Rightarrow$ CASE 1: $T(n) = \Theta(n^2)$.
- $T(n) = 4T(n/2) + n^2$
  $n^{\log_b a} = n^2 = n^2 \lg^0 n \Rightarrow$ CASE 2: $T(n) = \Theta(n^2 \lg n)$.
- $T(n) = 4T(n/2) + n^3$
  $n^{\log_b a} = n^2 \ll n^3 \Rightarrow$ CASE 3: $T(n) = \Theta(n^3)$.
- $T(n) = 4T(n/2) + n^2/\lg n$
  Master method does not apply!
  Answer is $T(n) = \Theta(n^2 \lg\lg n)$. (Prove by substitution.)

More general (but more mathematically sophisicated) solution: ***Akra–Bazzi method***.