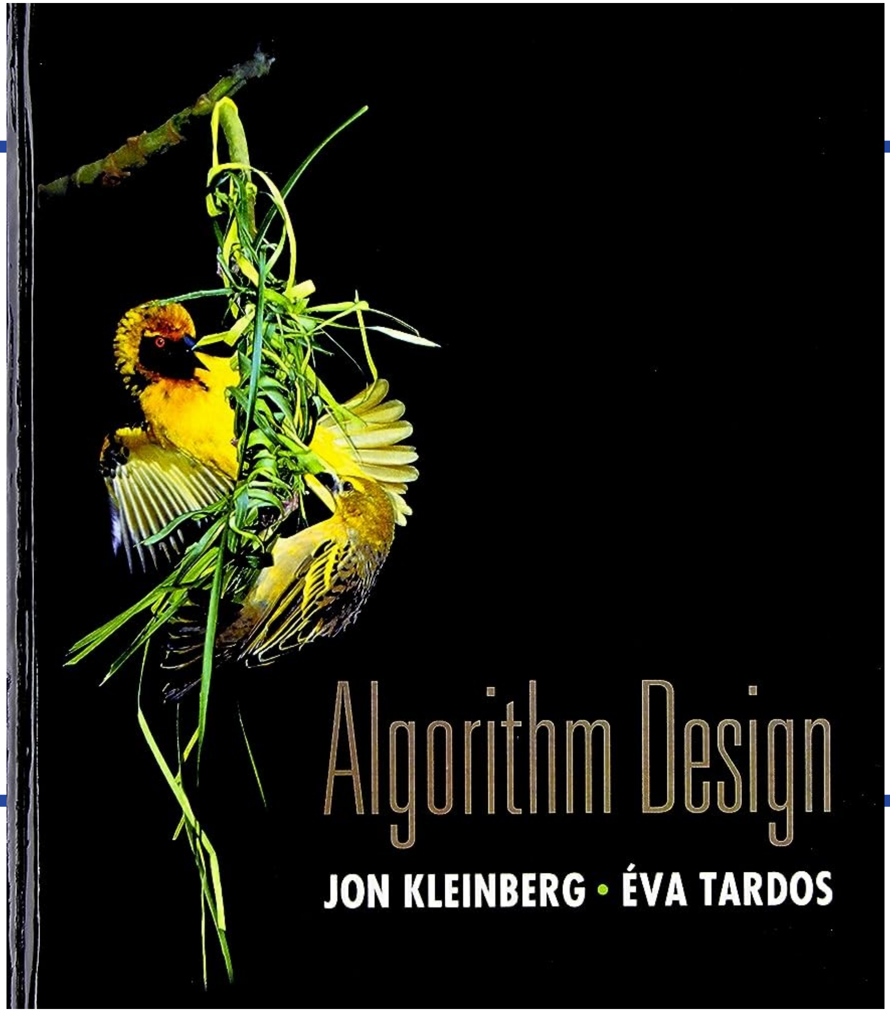
CS 310: Algorithms

Lecture 13

Instructor: Naveed Anwar Bhatti

Administrivia

Assignment 2 released: **Deadline 21st October**



Chapter 5:
Divide and Conquer

Master Theorem

- Use of Master Theorem (last time)
- Proof of Master Theorem (**today**)

What is Master Theorem?

Theorem

If $T(n) = aT\left(\frac{n}{b}\right) + f(n)$ (for constants **a**>0, **b**> 1)

Then let $T(n)$ has the following asymptotic bounds:

1. If $f(n) = O\left(n^{\log_b a - \epsilon}\right)$ for some constant $\epsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$
2. If $f(n) = \Theta\left(n^{\log_b a}\right)$, then $T(n) = \Theta\left(n^{\log_b a}\, lg\, n\right)$
3. If $f(n) = \Omega\left(n^{\log_b a + \epsilon}\right)$ for some constant $\epsilon > 0$, and if $af(n/b) \leq cf(n)$, for some constant $c < 1$ and all sufficiently large n, then $T(n) = \Theta(f(n))$

Master Method – Example 1

- Using Master Theorem, find the asymptotic bounds of:

$$T(n) = 9T\left(\frac{n}{3}\right) + n$$

Master Theorem

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

1. If $f(n) = O\left(n^{\log_b a - \epsilon}\right)$ for some constant $\epsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$
2. If $f(n) = \Theta\left(n^{\log_b a}\right)$, then $T(n) = \Theta\left(n^{\log_b a}\, lg\, n\right)$
3. If $f(n) = \Omega\left(n^{\log_b a + \epsilon}\right)$ for some constant $\epsilon > 0$, and if $af(n/b) \leq cf(n)$, for some constant $c < 1$ and all sufficiently large n, then $T(n) = \Theta(f(n))$

Master Method – Example 2

- Using Master Theorem, find the asymptotic bounds of:

$$T(n) = T\left(\frac{2n}{3}\right) + 1$$

Master Theorem

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

1. If $f(n) = O\left(n^{\log_b a-\epsilon}\right)$ for some constant $\epsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$
2. If $f(n) = \Theta\left(n^{\log_b a}\right)$, then $T(n) = \Theta\left(n^{\log_b a}\, lg\, n\right)$
3. If $f(n) = \Omega\left(n^{\log_b a+\epsilon}\right)$ for some constant $\epsilon > 0$, and if $af(n/b) \leq cf(n)$, for some constant $c < 1$ and all sufficiently large n, then $T(n) = \Theta(f(n))$

Master Method – Example 3

- Using Master Theorem, find the asymptotic bounds of:

$$T(n) = 3T\left(\frac{n}{4}\right) + nlgn$$

Master Theorem

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

1. If $f(n) = O\left(n^{\log_b a-\epsilon}\right)$ for some constant $\epsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$
2. If $f(n) = \Theta\left(n^{\log_b a}\right)$, then $T(n) = \Theta\left(n^{\log_b a}\, lg\, n\right)$
3. If $f(n) = \Omega\left(n^{\log_b a+\epsilon}\right)$ for some constant $\epsilon > 0$, and if $af(n/b) \leq cf(n)$, for some constant $c < 1$ and all sufficiently large n, then $T(n) = \Theta(f(n))$

Master Method – Example 4

- Using Master Theorem, find the asymptotic bounds of:

$$T(n) = 2T\left(\frac{n}{2}\right) + \Theta(n)$$

Master Theorem

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

1. If $f(n) = O\left(n^{\log_b a-\epsilon}\right)$ for some constant $\epsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$
2. If $f(n) = \Theta\left(n^{\log_b a}\right)$, then $T(n) = \Theta\left(n^{\log_b a}\, lg\, n\right)$
3. If $f(n) = \Omega\left(n^{\log_b a+\epsilon}\right)$ for some constant $\epsilon > 0$, and if $af(n/b) \leq cf(n)$, for some constant $c < 1$ and all sufficiently large n, then $T(n) = \Theta(f(n))$

Master Method – Example 5

- Using Master Theorem, find the asymptotic bounds of:

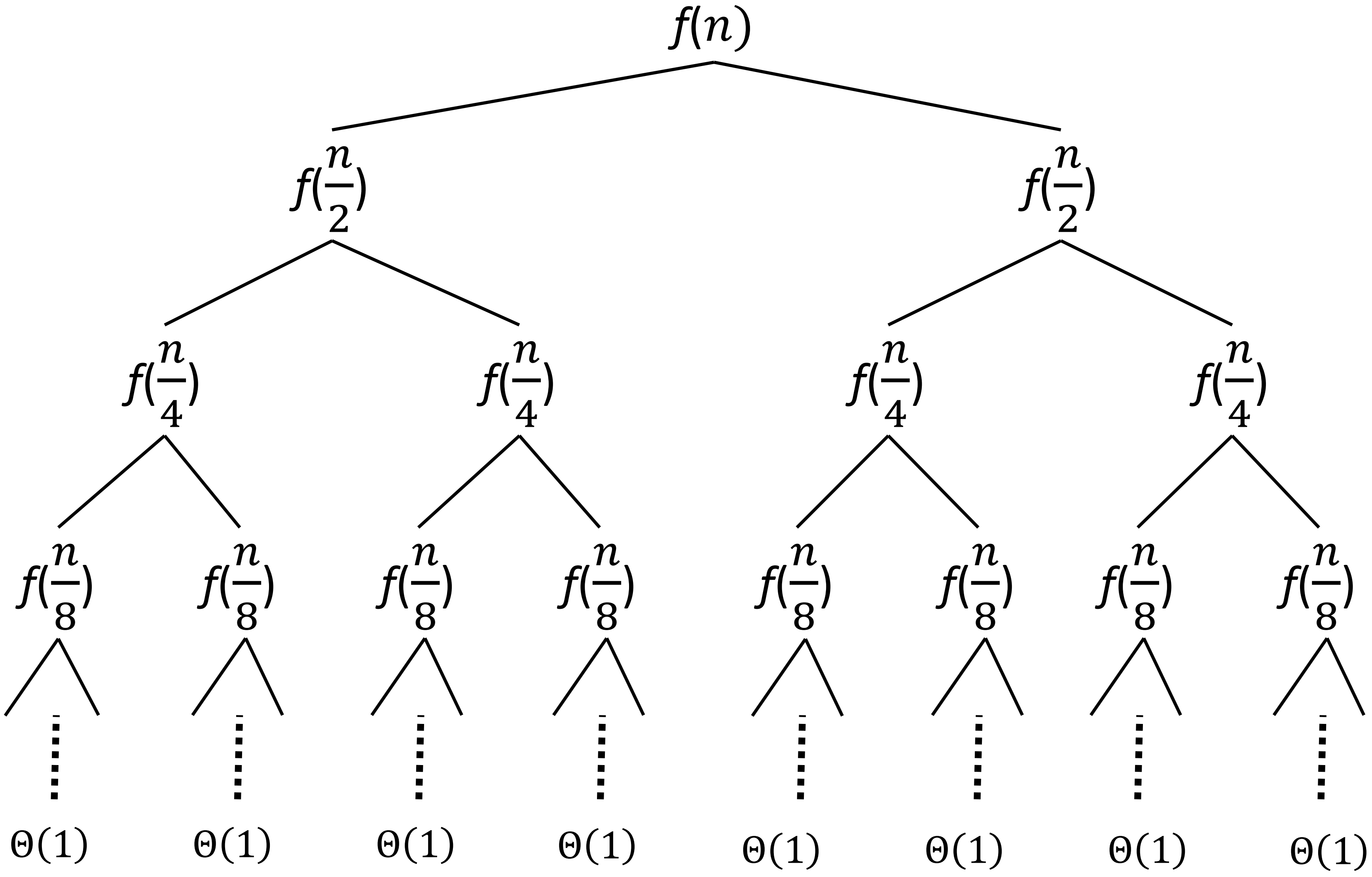
$$T(n) = 8T\left(\frac{n}{2}\right) + \Theta(n^2)$$

Master Theorem

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

1. If $f(n) = O\left(n^{\log_b a-\epsilon}\right)$ for some constant $\epsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$
2. If $f(n) = \Theta\left(n^{\log_b a}\right)$, then $T(n) = \Theta\left(n^{\log_b a}\, lg\, n\right)$
3. If $f(n) = \Omega\left(n^{\log_b a+\epsilon}\right)$ for some constant $\epsilon > 0$, and if $af(n/b) \leq cf(n)$, for some constant $c < 1$ and all sufficiently large n, then $T(n) = \Theta(f(n))$

Master Theorem – Proof (Merge-Sort Example)



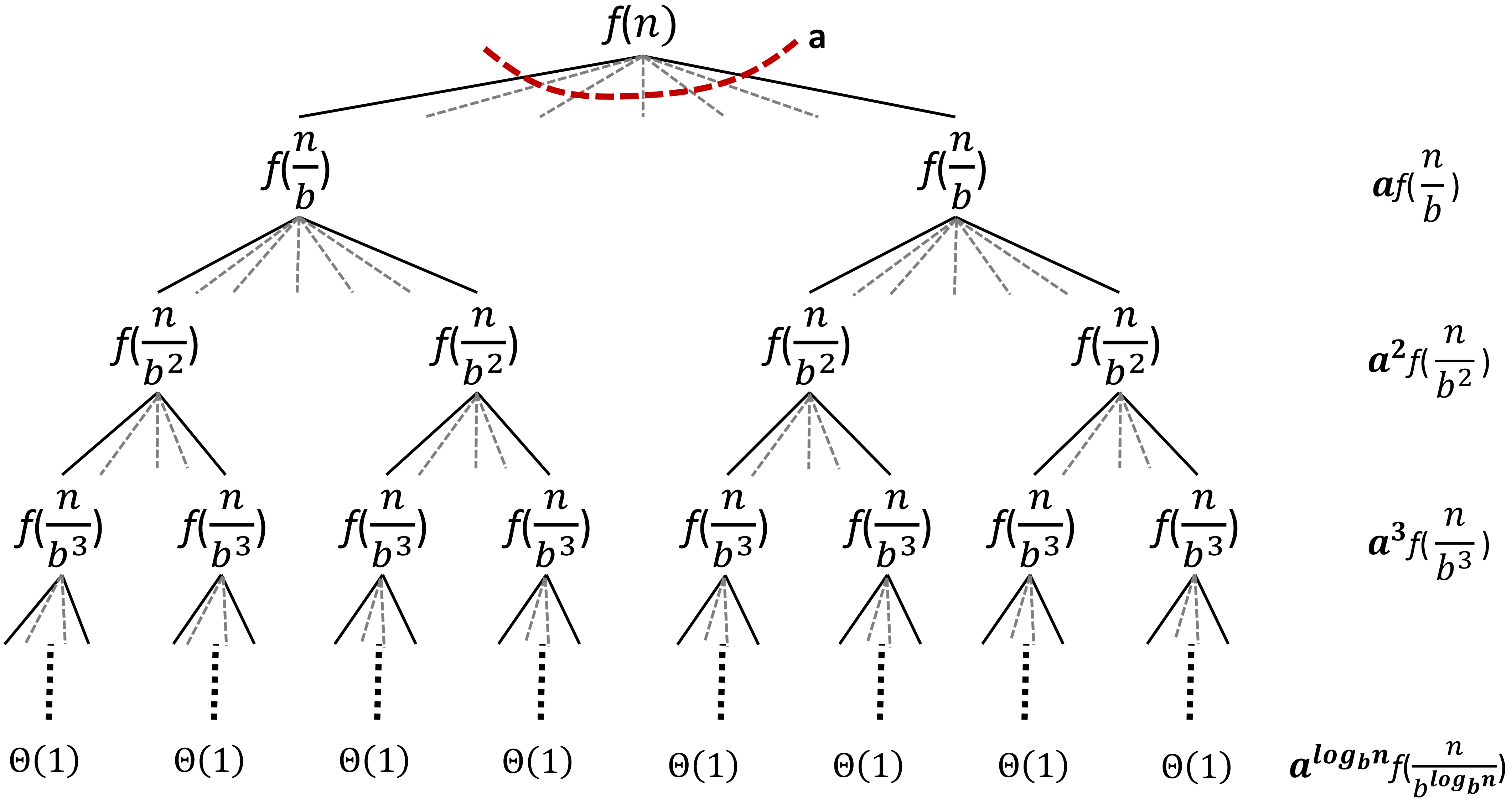
$\mathbf{2}\, f(\frac{n}{2})$ $\quad \mathbf{2^1} f(\frac{n}{\mathbf{2^1}})$

$\mathbf{4}\, f(\frac{n}{4})$ $\quad \mathbf{2^2} f(\frac{n}{\mathbf{2^2}})$

$\mathbf{8}\, f(\frac{n}{8})$ $\quad \mathbf{2^3} f(\frac{n}{\mathbf{2^3}})$

$\mathbf{2^{log_2 n}}\, f(\frac{n}{2^{\mathbf{log_2 n}}})$

Master Theorem - Proof



$$\text{Total cost is}, T(n) = f(n) + \boldsymbol{a} f(\frac{n}{b}) + \boldsymbol{a^2} f(\frac{n}{b^2}) + \boldsymbol{a^3} f(\frac{n}{b^3}) + \cdots\cdots + \boldsymbol{a^{log_b n}} f(\frac{n}{b^{\boldsymbol{log_b n}}})$$

$$\text{Total cost is}, T(n) = \sum_{j=0}^{\log_b n} a^j f(\frac{n}{b^j})$$

$$\text{Total cost is}, T(n) = \boldsymbol{a^{log_b n}} f(\frac{n}{b^{\boldsymbol{log_b n}}}) + \sum_{j=0}^{\log_b n-1} a^j f(\frac{n}{b^j})$$

$$\text{Total cost is}, T(n) = \boldsymbol{a^{log_b n}} f(1) + \sum_{j=0}^{\log_b n-1} a^j f(\frac{n}{b^j})$$

$$\text{Total cost is}, T(n) = \boldsymbol{a^{log_b n}} + \sum_{j=0}^{\log_b n-1} a^j f(\frac{n}{b^j})$$

$$\boldsymbol{log_b n = x}$$

$$\boldsymbol{b^x = n}$$

$$\boldsymbol{b^{log_b n} = n}$$

Total cost is, $T(n) = \boldsymbol{a^{\log_b n}} + \sum_{j=0}^{\log_b n-1} a^j f(\frac{n}{b^j})$

$$\boxed{\text{Total cost is, } T(n) = \boldsymbol{n^{\log_b a}} + \sum_{j=0}^{\log_b n-1} a^j f(\frac{n}{b^j})}$$

$\boldsymbol{\log_b n}$

$\boldsymbol{\log_k n} = \frac{\boldsymbol{\log_t n}}{\boldsymbol{\log_t k}}$

$\boldsymbol{\log_b n} = \frac{\boldsymbol{\log_a n}}{\boldsymbol{\log_a b}}$

$\boldsymbol{\log_k s} = \frac{\boldsymbol{1}}{\boldsymbol{\log_s k}}$

$\boldsymbol{\log_b n} = \boldsymbol{\log_b a \ \log_a n}$

$\boldsymbol{k \log x = \log x^k}$

$\boldsymbol{\log_b n} = \boldsymbol{\log_a n^{\log_b a}}$

$\boldsymbol{\log_b n \log_a a} = \boldsymbol{\log_a n^{\log_b a}}$

$\boldsymbol{\log_a a^{\log_b n}} = \boldsymbol{\log_a n^{\log_b a}}$

$\boldsymbol{a^{\log_b n}} = \boldsymbol{n^{\log_b a}}$

Case 1: $f(n) = \left(n^{\log_b a - \epsilon}\right)$

$$\text{Total cost is}, T(n) = \boldsymbol{n^{log_b a}} + \sum_{j=0}^{\log_b n-1} a^j f(\frac{n}{b^j})$$

$$= \boldsymbol{n^{log_b a}} + \sum_{j=0}^{\log_b n-1} a^j (\frac{n}{b^j})^{\log_b a-\epsilon}$$

$$= \boldsymbol{n^{log_b a}} + \boldsymbol{n}^{\log_b a-\epsilon} \sum_{j=0}^{\log_b n-1} a^j (\frac{1}{b^{j(\log_b a-\epsilon)}})$$

$$= \boldsymbol{n^{log_b a}} + \boldsymbol{n}^{\log_b a-\epsilon} \sum_{j=0}^{\log_b n-1} a^j (\frac{1}{b^{(\log_b a^j - j\epsilon)}})$$

$$= \boldsymbol{n^{log_b a}} + \boldsymbol{n}^{\log_b a-\epsilon} \sum_{j=0}^{\log_b n-1} a^j (\frac{b^{j\epsilon}}{b^{\log_b a^j}})$$

Case 1: $f(n) = \left(n^{\log_b a - \epsilon}\right)$

$$\text{Total cost is, } T(n) = \boldsymbol{n^{log_b a}} + \sum_{j=0}^{\log_b n - 1} a^j f\left(\frac{n}{b^j}\right)$$

$$= \boldsymbol{n^{log_b a}} + \boldsymbol{n}^{\log_b a - \epsilon} \sum_{j=0}^{\log_b n - 1} a^j \left(\frac{b^{j\epsilon}}{b^{\log_b a^j}}\right)$$

$$= \boldsymbol{n^{log_b a}} + \boldsymbol{n}^{\log_b a - \epsilon} \sum_{j=0}^{\log_b n - 1} a^j \left(\frac{b^{j\epsilon}}{a^j}\right)$$

$$= \boldsymbol{n^{log_b a}} + \frac{\boldsymbol{n}^{\log_b a}}{\boldsymbol{n}^{\epsilon}} \sum_{j=0}^{\log_b n - 1} b^{j\epsilon}$$

$$= \boldsymbol{n^{log_b a}}\left(1 + \frac{1}{\boldsymbol{n}^{\epsilon}} \sum_{j=0}^{\log_b n - 1} b^{j\epsilon}\right)$$

Case 1: $f(n) = \left(n^{\log_b a - \epsilon}\right)$

$$\text{Total cost is}, T(n) = \boldsymbol{n^{log_b a}} + \sum_{j=0}^{\log_b n - 1} a^j f(\frac{n}{b^j})$$

$$= \boldsymbol{n^{log_b a}}(1 + \frac{1}{\boldsymbol{n^{\epsilon}}} \sum_{j=0}^{\log_b n - 1} b^{j\epsilon})$$

To solve this, we'll sum the terms for every value of j from 0 **to** $\log_b n - 1$

$$= \boldsymbol{n^{log_b a}}(1 + \frac{1}{\boldsymbol{n^{\epsilon}}}\left(\frac{b^{\epsilon \log_b n} - 1}{b^{\epsilon} - 1}\right))$$

When j=0, $b^{j\epsilon}$ =*1*
When j=1, $b^{j\epsilon}$ = b^{ϵ}
When j=2, $b^{j\epsilon}$ = $b^{2\epsilon}$

$$\sum_{j=0}^{n} k^j = \left(\frac{k^{n+1} - 1}{r - 1}\right)$$

$$= \boldsymbol{n^{log_b a}}(1 + \frac{1}{\boldsymbol{n^{\epsilon}}}\left(\frac{b^{\log_b n^{\epsilon}} - 1}{b^{\epsilon} - 1}\right))$$

$$= \boldsymbol{n^{log_b a}}\left(1 + \frac{1}{\boldsymbol{n^{\epsilon}}}\left(\frac{n^{\epsilon} - 1}{b^{\epsilon} - 1}\right)\right) = \Theta(\boldsymbol{n^{log_b a}})$$

Hence proved !!!!!

Case 2: $f(n) = \left(n^{\log_b a}\right)$

$$\text{Total cost is}, T(n) = \boldsymbol{n^{log_b a}} + \sum_{j=0}^{\log_b n-1} a^j f(\frac{n}{b^j})$$

$$= \boldsymbol{n^{log_b a}}(1 + \frac{1}{\boldsymbol{n^{\epsilon}}} \sum_{j=0}^{\log_b n-1} b^{j\epsilon})$$

$$= \boldsymbol{n^{log_b a}}(1 + \sum_{j=0}^{\log_b n-1} 1)$$

$$= \boldsymbol{n^{log_b a}}(1 + \log_b(n - 1 \ + 1))$$

$$= \boldsymbol{n^{log_b a}}(1 + \log_b n\ )$$

$$= \Theta(\boldsymbol{n^{log_b a}} \log_b n\ )$$ Hence proved !!!!!

Case 3: $f(n) = \left(n^{\log_b a+\epsilon}\right)$

$$\text{Total cost is, } T(n) = \boldsymbol{n^{log_b a}} + \sum_{j=0}^{\log_b n-1} a^j f\left(\frac{n}{b^j}\right)$$

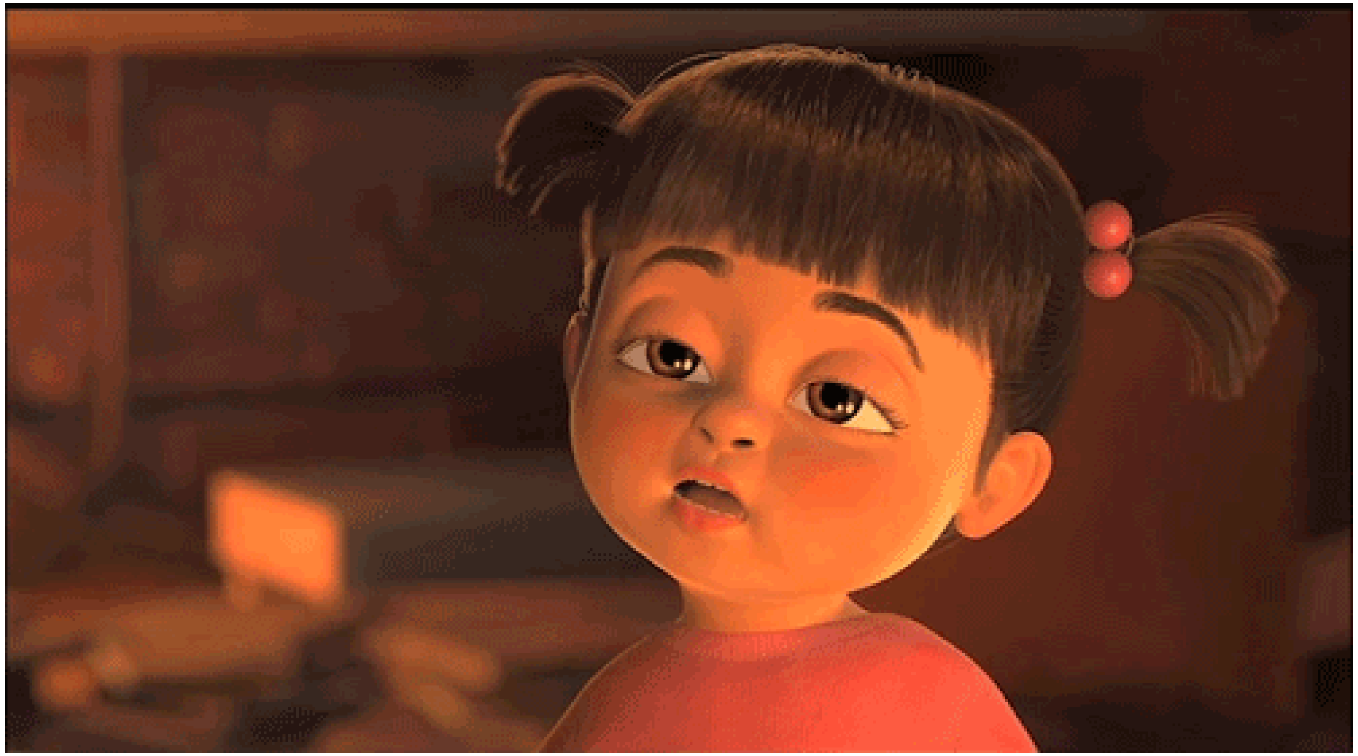
$$= \boldsymbol{n^{log_b a}} + \boldsymbol{n}^{\log_b a+\epsilon} \sum_{j=0}^{\log_b n-1} \left(\frac{1}{b^{j\epsilon}}\right)$$

$$= \boldsymbol{n^{log_b a}} + \boldsymbol{n}^{\log_b a+\epsilon} \left(\frac{1}{1-b^{\epsilon}}\right)$$

$$= \left(\frac{n^{log_b a} + \boldsymbol{n^{log_b a+\epsilon}} - b^{\epsilon} n^{log_b a}}{1-b^{\epsilon}}\right)$$

$$= \Theta(\boldsymbol{n^{log_b a+\epsilon}})$$

Thanks a lot



If you are taking a Nap, **wake up**.......Lecture Over