Analyzing
Divide and Conquer

Substitution Method

1. Adivinar la forma de la solución
2. Utilizar inducción matemática para demostrar que esta solución funciona

Substitution Method

Ejemplo de Mergesort:

$$T(n) = 2T(n/2) + n$$

Adivinamos que la solución es:

$$T(n) = O(n \lg(n))$$

Substitution Method

Ejemplo de Mergesort:

$$T(n) = 2T(n/2) + n$$

Adivinamos que la solución es:

$$T(n) = O(n \lg(n))$$

Comprobamos que:

$$T(n) <= cn \lg(n)$$

Substitution Method

Asumimos que la condición se mantiene para todo $m < n$. Donde m = $n/2$

$$T(n) \leq 2(c \lfloor n/2 \rfloor \lg(\lfloor n/2 \rfloor)) + n$$

Substitution Method

$$\begin{aligned}
T(n) &\leq 2(c\lfloor n/2 \rfloor \lg(\lfloor n/2 \rfloor)) + n \\
&\leq cn \lg(n/2) + n \\
&= cn \lg n - cn \lg 2 + n \\
&= cn \lg n - cn + n \\
&\leq cn \lg n \ ,
\end{aligned}$$

Substitution Method

$$
\begin{aligned}
T(n) &\le 2(c\lfloor n/2\rfloor \lg(\lfloor n/2\rfloor)) + n \\
&\le cn\lg(n/2) + n \\
&= cn\lg n - cn\lg 2 + n \\
&= cn\lg n - cn + n \\
&\le cn\lg n\ ,
\end{aligned}
$$

Substitution Method

Caso base:

$$n = 2,3$$

Remplazando:

$$T(2) = 2 \lg(2) = 2$$

$$T(3) = 3 \lg(3) = 5\sim$$

Substitution Method

Hipotesis inductiva:

$$T(n) <= cn \lg(n)$$

Donde hay un c >= 1 que cumple con la condición.

Remplazando:

C = 2

$$T(2) <= 2 * (2 \lg(2))$$

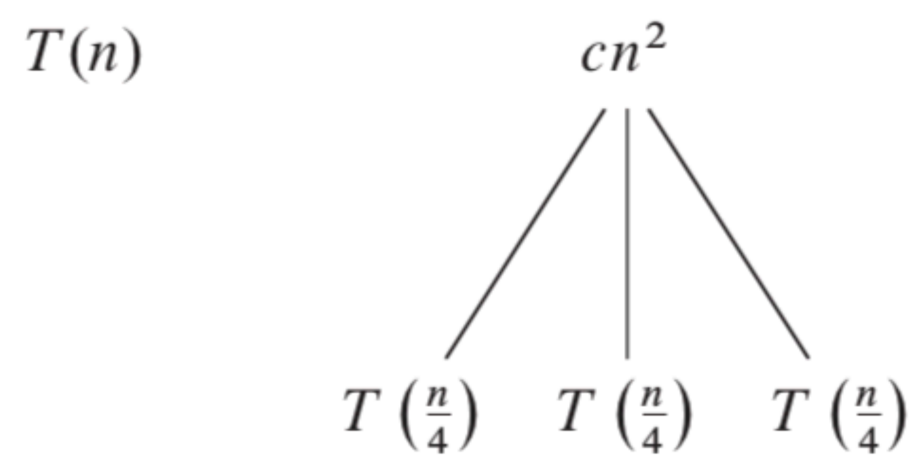
$$T(3) <= 2 * (3 \lg(3))$$

Recursion tree Method

$$T(n) = 3T(\lfloor n/4 \rfloor) + \Theta(n^2).$$

Recursion tree Method

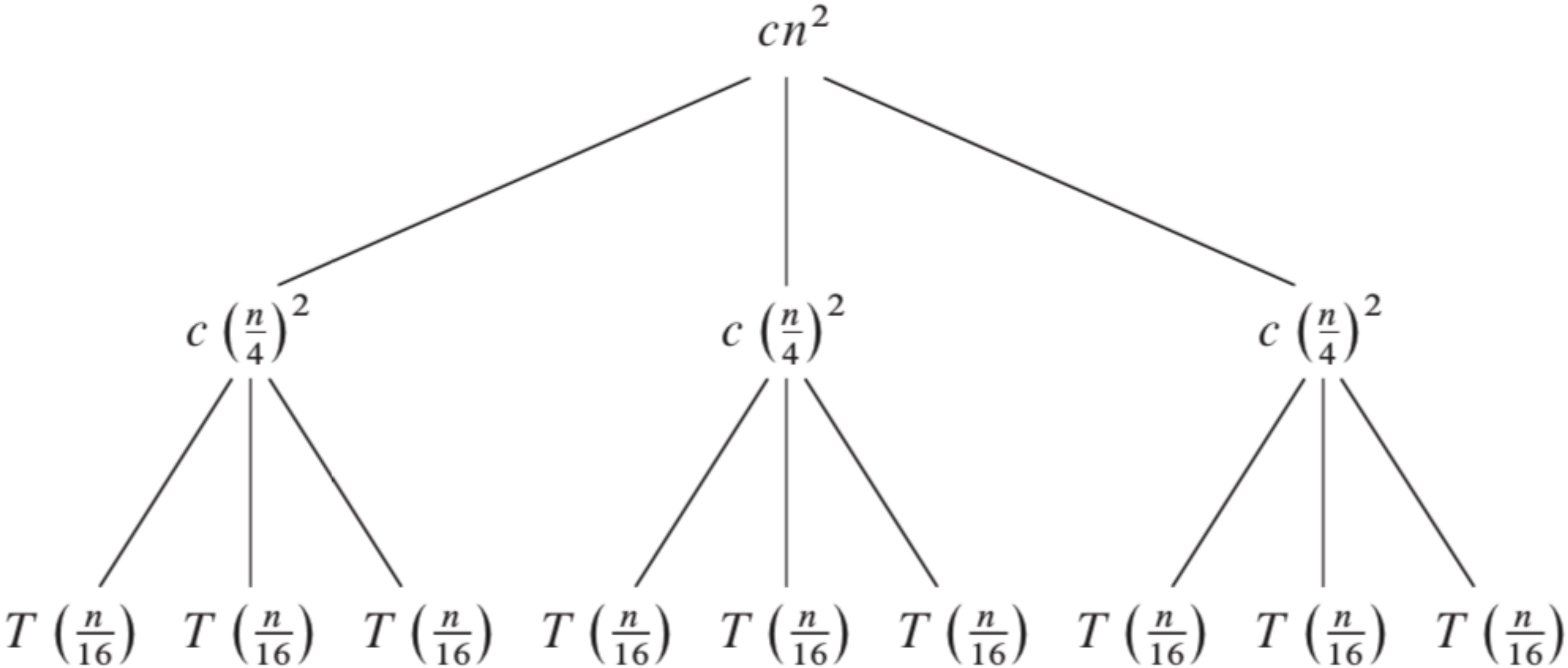
$$T(n) = 3T(\lfloor n/4 \rfloor) + \Theta(n^2).$$



(a) (b)

Recursion tree Method

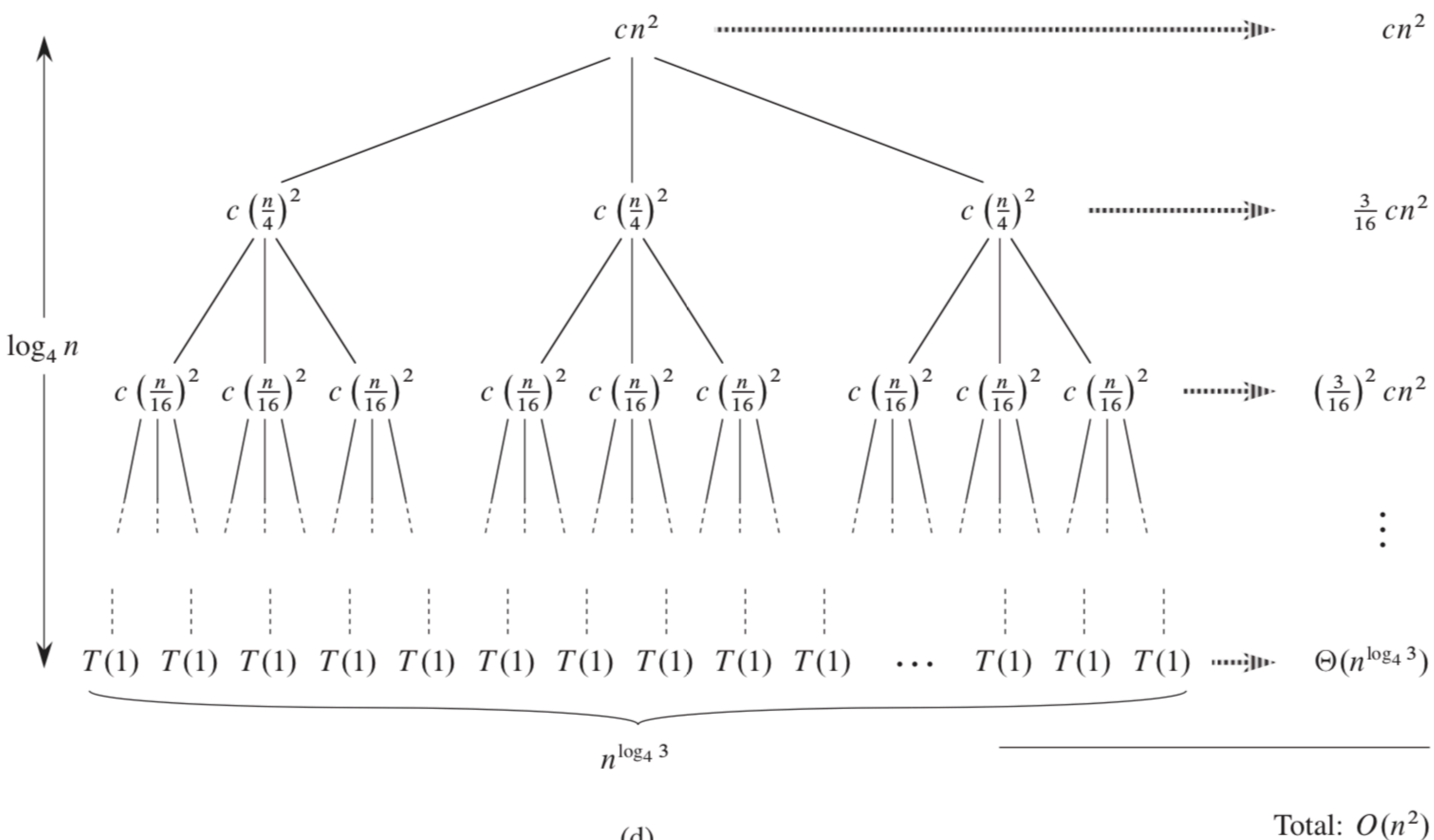
$$T(n) = 3T(\lfloor n/4 \rfloor) + \Theta(n^2).$$



(c)

Recursion tree Method

$$T(n) = 3T(\lfloor n/4 \rfloor) + \Theta(n^2).$$



(d)

Recursion tree Method

$$T(n) = 3T(\lfloor n/4 \rfloor) + \Theta(n^2).$$

$$\begin{aligned} T(n) &= cn^2 + \frac{3}{16}cn^2 + \left(\frac{3}{16}\right)^2 cn^2 + \cdots + \left(\frac{3}{16}\right)^{\log_4 n - 1} cn^2 + \Theta(n^{\log_4 3}) \\ &= \sum_{i=0}^{\log_4 n - 1} \left(\frac{3}{16}\right)^i cn^2 + \Theta(n^{\log_4 3}) \end{aligned}$$

Recursion tree Method

$$T(n) = 3T(\lfloor n/4 \rfloor) + \Theta(n^2).$$

$$\begin{aligned} T(n) &= cn^2 + \frac{3}{16}cn^2 + \left(\frac{3}{16}\right)^2 cn^2 + \cdots + \left(\frac{3}{16}\right)^{\log_4 n-1} cn^2 + \Theta(n^{\log_4 3}) \\ &= \sum_{i=0}^{\log_4 n-1} \left(\frac{3}{16}\right)^i cn^2 + \Theta(n^{\log_4 3}) \end{aligned}$$

Recursion tree Method

$$T(n) = 3T(\lfloor n/4 \rfloor) + \Theta(n^2).$$

$$\begin{aligned} T(n) &= cn^2 + \frac{3}{16}cn^2 + \left(\frac{3}{16}\right)^2 cn^2 + \cdots + \left(\frac{3}{16}\right)^{\log_4 n-1} cn^2 + \Theta(n^{\log_4 3}) \\ &= \sum_{i=0}^{\log_4 n-1} \left(\frac{3}{16}\right)^i cn^2 + \Theta(n^{\log_4 3}) \end{aligned}$$

Recursion tree Method

$$T(n) = 3T(\lfloor n/4 \rfloor) + \Theta(n^2).$$

$$\begin{aligned}
T(n) &= \sum_{i=0}^{\log_4 n - 1} \left(\frac{3}{16}\right)^i cn^2 + \Theta(n^{\log_4 3}) \\
&< \sum_{i=0}^{\infty} \left(\frac{3}{16}\right)^i cn^2 + \Theta(n^{\log_4 3}) \\
&= \frac{1}{1-(3/16)} cn^2 + \Theta(n^{\log_4 3}) \\
&= \frac{16}{13} cn^2 + \Theta(n^{\log_4 3}) \\
&= O(n^2) .
\end{aligned}$$

Recursion tree Method

Aplicando substitución:

$$
\begin{aligned}
T(n) &\leq 3T(\lfloor n/4 \rfloor) + cn^2 \\
&\leq 3d \lfloor n/4 \rfloor^2 + cn^2 \\
&\leq 3d(n/4)^2 + cn^2 \\
&= \frac{3}{16} dn^2 + cn^2 \\
&\leq dn^2 ,
\end{aligned}
$$

Recursion tree Method

Aplicando substitución donde

n = n^2:

$$
\begin{aligned}
T(n) &\leq 3T(\lfloor n/4 \rfloor) + cn^2 \\
&\leq 3d \lfloor n/4 \rfloor^2 + cn^2 \\
&\leq 3d(n/4)^2 + cn^2 \\
&= \frac{3}{16} dn^2 + cn^2 \\
&\leq dn^2 ,
\end{aligned}
$$

Master Method

Receta mestra para resolver recurrencias de la forma:

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

n= tamaño del problema
a= numero de subproblemas que surgen
n/b = tamaño de subproblemas
f(n) = costo de dividir y unir los subproblemas

Master Method

Theorem 4.1 (Master theorem)
Let $a \geq 1$ and $b > 1$ be constants, let $f(n)$ be a function, and let $T(n)$ be defined on the nonnegative integers by the recurrence

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

where we interpret n/b to mean either $\lfloor n/b \rfloor$ or $\lceil n/b \rceil$. Then $T(n)$ has the following asymptotic bounds:

1. If $f(n) = O(n^{\log_b a - \epsilon})$ for some constant $\epsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$.
2. If $f(n) = \Theta(n^{\log_b a})$, then $T(n) = \Theta(n^{\log_b a} \lg n)$.
3. If $f(n) = \Omega(n^{\log_b a + \epsilon})$ 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

1. The running time is dominated by the cost at the leaves:

If $f(n) = O(n^{\log_b(a) - \varepsilon})$, then $T(n) = \Theta(n^{\log_b(a)})$

for an $\varepsilon > 0$

2. The running time is evenly distributed throughout the tree:

If $f(n) = \Theta(n^{\log_b(a)})$, then $T(n) = \Theta(n^{\log_b(a)} \log(n))$

3. The running time is dominated by the cost at the root:

If $f(n) = \Omega(n^{\log_b(a) + \varepsilon})$, then $T(n) = \Theta(f(n))$

for an $\varepsilon > 0$

If f(n) satisfies the regularity condition:
$af(n/b) <= cf(n)$ where $c < 1$ (this always holds for polynomials)
Because of this condition, the Master Method cannot solve every recurrence of the given form.

Master Method

$$T(n) = 9T(n/3) + n \; .$$

$$T(n) = 3T(n/4) + n \lg n \; ,$$

$$T(n) = T(2n/3) + 1,$$

Master Method

Theorem 4.1 (Master theorem)
Let $a \geq 1$ and $b > 1$ be constants, let $f(n)$ be a function, and let $T(n)$ be defined on the nonnegative integers by the recurrence

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

where we interpret n/b to mean either $\lfloor n/b \rfloor$ or $\lceil n/b \rceil$. Then $T(n)$ has the following asymptotic bounds:

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