Section 8.17
Divide-and-Conquer Recurrence Relations

A Review of Recurrence Relations for Divide-and-Conquer Algorithms

- Recall the recurrence relations that describe the number of operations used by some divide-and-conquer algorithms (Sections 8.13 and 8.14)

- Finding the minimum of a sequence:

$$T(1) = 2$$

$$T(n) = 2T(n/2) + 8$$

A Review of Recurrence Relations for Divide-and-Conquer Algorithms

- Merge Sort:

$$T(1) = 2$$

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

- Note: $T(n) = 2T(n/2) + \Theta(n)$ means $T(n) = 2T(n/2) + f(n)$ for some function $f(n)$ that is $\Theta(n)$

A Review of Recurrence Relations for Divide-and-Conquer Algorithms

- Binary Search:

$$T(1) = 3$$

$$T(n) = T(n/2) + 9$$

Divide-and-Conquer Recurrence Relation

- Many recurrence relations counting the number of operations for divide-and-conquer algorithms are of the form:

$$T(1) = c$$

$$T(n) = aT(n/b) + \Theta(n^d)$$

where $T(n) = aT(n/b) + \Theta(n^d)$ means $T(n) = aT(n/b) + f(n)$ for some function $f(n)$ that is $\Theta(n^d)$

The Master Theorem

Consider a recurrence relation and initial condition of the following form where a, b, c, and d are constants:

$$T(1) = c$$
$$T(n) = aT(n/b) + \Theta(n^d)$$

1. If $a/b^d = 1$, then $T(n)$ is $\Theta\left(n^d log(n)\right)$
2. If $a/b^d < 1$, then $T(n)$ is $\Theta(n^d)$
3. If $a/b^d > 1$, then $T(n)$ is $\Theta(n^{log_b(a)})$

Master Theorem Examples

Example 1: The number of operations used by divide-and-conquer algorithm for finding the minimum of a sequence of length n is:

$$T(1) = 2$$
$$T(n) = 2T(n/2) + 8$$

Note 8 is $\Theta\left(n^0\right)$
$a = 2, b = 2, d = 0$

$$a/b^d = 2/\left(2^0\right) = 2 > 1$$

$T(n)$ is $\Theta\left(n^{log_2(2)}\right)$
$T(n)$ is $\Theta(n)$

Master Theorem Examples

Example 2: The number of operations used by Merge sort on a sequence of length n is:

$$T(1) = 2$$
$$T(n) = 2T(n/2) + \Theta(n)$$

$a = 2, b = 2, d = 1$

$$a/b^d = 2/(2^1) = 1$$

$T(n)$ is $\Theta(n \cdot log(n))$

Master Theorem Examples

Example 3: The number of operations used by binary search on a sequence of length n is:

$$T(1) = 3$$
$$T(n) = T(n/2) + 9$$

$a = 1, b = 2, d = 0$

$$a/b^d = 1/\left(2^0\right) = 1$$

$T(n)$ is $\Theta(\ log_2(n))$

Building a Recursion Tree

- Example: Consider an algorithm that uses the following number of operations for inputs of size n:

$$T(1) = 1$$

$$T(n) = 3T(n/2) + n^5$$

Build a tree that describes the calculation of $T(n)$

Building a Recursion Tree

$T(1) = 1$

$T(n) = 3T(n/2) + n^5$

$T(n)$

Building a Recursion Tree

$$T(1) = 1$$

$$T(n) = 3T(n/2) + n^5$$

$$T(n/2) + T(n/2) + T(n/2) + n^5$$

Building a Recursion Tree

$T(1) = 1$

$T(n) = 3T(n/2) + n^5$

n^5

$T(n/2)$ $T(n/2)$ $T(n/2)$

Building a Recursion Tree

$T(1) = 1$

$T(n) = 3T(n/2) + n^5$

n^5

$3T(n/4) + (n/2)^5$ $3T(n/4) + (n/2)^5$ $3T(n/4) + (n/2)^5$

Building a Recursion Tree

$T(1) = 1$

$T(n) = 3T(n/2) + n^5$

n^5

$(n/2)^5$ $(n/2)^5$ $(n/2)^5$

$T(n/4)$ $T(n/4)$ $T(n/4)$ $T(n/4)$ $T(n/4)$ $T(n/4)$ $T(n/4)$ $T(n/4)$ $T(n/4)$

Building a Recursion Tree

$T(1) = 1$

$T(n) = 3T(n/2) + n^5$

n^5

$(n/2)^5$ $(n/2)^5$ $(n/2)^5$

$(n/4)^5$ $(n/4)^5$ $(n/4)^5$ $(n/4)^5$ $(n/4)^5$ $(n/4)^5$ $(n/4)^5$ $(n/4)^5$ $(n/4)^5$

$\vdots$ $\vdots$ $\vdots$

Building a Recursion Tree

$T(1) = 1$

$T(n) = 3T(n/2) + n^5$

n^5

$(n/2)^5$ $(n/2)^5$ $(n/2)^5$

$(n/4)^5$ $(n/4)^5$ $(n/4)^5$ $(n/4)^5$ $(n/4)^5$ $(n/4)^5$ $(n/4)^5$ $(n/4)^5$ $(n/4)^5$

$\vdots$ $\vdots$ $\vdots$

$\cdots$ 1 1 1 1 1 1 1 1 1 $\cdots$

Building a Recursion Tree

$T(1) = 1$

$T(n) = 3T(n/2) + n^5$

n^5 $\qquad$ $3^0 \cdot \left(n/2^0\right)^5$

$(n/2)^5 \quad (n/2)^5 \quad (n/2)^5$ $\qquad$ $3^1 \cdot (n/2^1)^5$

$(n/4)^5 \quad (n/4)^5 \quad (n/4)^5 \quad (n/4)^5 \quad (n/4)^5 \quad (n/4)^5 \quad (n/4)^5 \quad (n/4)^5 \quad (n/4)^5$ $\qquad$ $3^2 \cdot \left(n/2^2\right)^5$

$\vdots \quad \vdots \quad \vdots$

$\cdots \quad 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1 \quad \cdots$ $\qquad$ $3^L \cdot \left(n/2^L\right)^5$

$L = log_2(n)$

Building a Recursion Tree

$T(1) = 1$

$T(n) = 3T(n/2) + n^5$

n^5 $\qquad 3^0 \cdot (n/2^0)^5$

$(n/2)^5 \quad (n/2)^5 \quad (n/2)^5$ $\qquad 3^1 \cdot (n/2^1)^5$

$(n/4)^5 \quad (n/4)^5 \quad (n/4)^5 \quad (n/4)^5 \quad (n/4)^5 \quad (n/4)^5 \quad (n/4)^5 \quad (n/4)^5 \quad (n/4)^5$ $\qquad 3^2 \cdot (n/2^2)^5$

$\vdots \quad \vdots \quad \vdots$

$\cdots \quad 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1 \quad \cdots$ $\qquad 3^L \cdot (n/2^L)^5$

$L = log_2(n)$

$$T(n) = \sum_{i=0}^{log_2(n)} 3^i \cdot (n/2^i)^5$$

Analyzing the Recursion Tree

$$T(1) = 1$$

$$T(n) = 3T(n/2) + n^5$$

$$T(n) \; = \sum_{i=0}^{log_2(n)} 3^i \cdot \left(n/2^i\right)^5$$

Analyzing the Recursion Tree

$T(1) = 1$

$T(n) = 3T(n/2) + n^5$

$$T(n) = \sum_{i=0}^{log_2(n)} 3^i \cdot \left(n/2^i\right)^5$$

$$= \sum_{i=0}^{log_2(n)} 3^i \cdot n^5/2^{5i}$$

Analyzing the Recursion Tree

$T(1) = 1$

$T(n) = 3T(n/2) + n^5$

$$T(n) = \sum_{i=0}^{log_2(n)} 3^i \cdot \left(n/2^i\right)^5$$

$$= \sum_{i=0}^{log_2(n)} 3^i \cdot n^5/2^{5i}$$

$$= n^5 \cdot \sum_{i=0}^{log_2(n)} 3^i/2^{5i}$$

Analyzing the Recursion Tree

$$T(1) = 1$$

$$T(n) = 3T(n/2) + n^5$$

$$T(n) = \sum_{i=0}^{log_2(n)} 3^i \cdot \left(n/2^i\right)^5$$

$$= \sum_{i=0}^{log_2(n)} 3^i \cdot n^5/2^{5i}$$

$$= n^5 \cdot \sum_{i=0}^{log_2(n)} 3^i/2^{5i}$$

$$= n^5 \cdot \sum_{i=0}^{log_2(n)} \left(3/2^5\right)^i$$

Analyzing the Recursion Tree

$T(1) = 1$

$T(n) = 3T(n/2) + n^5$

$$T(n) = \sum_{i=0}^{log_2(n)} 3^i \cdot \left(n/2^i\right)^5$$

$$= \sum_{i=0}^{log_2(n)} 3^i \cdot n^5/2^{5i}$$

$$= n^5 \cdot \sum_{i=0}^{log_2(n)} 3^i/2^{5i}$$

$$= n^5 \cdot \sum_{i=0}^{log_2(n)} \left(3/2^5\right)^i$$

Generalize:

$T(1) = 1$

$T(n) = aT(n/b) + n^d$

$$T(n) = n^d \cdot \sum_{i=0}^{log_b(n)} \left(a/b^d\right)^i$$

Solving the Recursion Tree

$T(1) = 1$

$T(n) = aT(n/b) + n^d$

$$T(n) = n^d \cdot \sum_{i=0}^{log_b(n)} \left(a/b^d\right)^i$$

Let $r = a/b^d$ (r is a constant determined by the form of the algorithm) and $m = log_b(n)$. There is a closed form solution for the sum of exponents:

Consider 3 cases:

1. $r = 1$
2. $r > 1$
3. $r < 1$

Solving the Recursion Tree

1. $a/b^d = 1$

$$
\begin{aligned}
T(n) &= n^d \cdot \sum_{i=0}^{log_b(n)} \left(a/b^d\right)^i \\
&= n^d \cdot \sum_{i=0}^{log_b(n)} 1^i \\
&= n^d(log_b(n) + 1\ )
\end{aligned}
$$

Hence $T(n)$ is $\Theta\left(n^d log(n)\right)$

Solving the Recursion Tree

$T(1) = 1$

$T(n) = aT(n/b) + n^d$

$$T(n) = n^d \cdot \sum_{i=0}^{log_b(n)} \left(a/b^d\right)^i$$

Let $r = a/b^d$ (r is a constant determined by the form of the algorithm) and $m = log_b(n)$. There is a closed form solution for the sum of exponents:

$$T(n) = n^d \cdot \sum_{i=0}^{m} r^i = \frac{r^{m+1} - 1}{r - 1}$$

When $r \neq 1$

Solving the Recursion Tree

2. $a/b^d < 1$

   Let $r = a/b^d$ and $m = log_b(n)$

   Use the closed form solution: $\sum_{i=0}^{m} r^i = \frac{r^{m+1}-1}{r-1}$

Solving the Recursion Tree

2. $a/b^d < 1$

   Let $r = a/b^d$ and $m = log_b(n)$

   Use the closed form solution: $\sum_{i=0}^{m} r^i = \frac{r^{m+1}-1}{r-1}$

$$T(n) \;= n^d \cdot \sum_{i=0}^{m} r^i$$

Solving the Recursion Tree

2. $a/b^d < 1$

   Let $r = a/b^d$ and $m = log_b(n)$

   Use the closed form solution: $\sum_{i=0}^{m} r^i = \frac{r^{m+1}-1}{r-1}$

$$T(n) = n^d \cdot \sum_{i=0}^{m} r^i$$

$$= n^d \cdot \frac{r^{m+1} - 1}{r - 1} \cdot \frac{-1}{-1}$$

Solving the Recursion Tree

2. $a/b^d < 1$

   Let $r = a/b^d$ and $m = log_b(n)$

   Use the closed form solution: $\sum_{i=0}^{m} r^i = \frac{r^{m+1}-1}{r-1}$

$$T(n) = n^d \cdot \sum_{i=0}^{m} r^i$$

$$= n^d \cdot \frac{r^{m+1} - 1}{r - 1} \cdot \frac{-1}{-1}$$

$$= n^d \cdot \frac{1 - r^{m+1}}{1 - r}$$

Solving the Recursion Tree

2. $a/b^d < 1$

   Let $r = a/b^d$ and $m = log_b(n)$

   Use the closed form solution: $\sum_{i=0}^{m} r^i = \frac{r^{m+1}-1}{r-1}$

$$T(n) = n^d \cdot \sum_{i=0}^{m} r^i$$

$$= n^d \cdot \frac{r^{m+1} - 1}{r - 1} \cdot \frac{-1}{-1}$$

$$= n^d \cdot \frac{1 - r^{m+1}}{1 - r}$$

$$\frac{1 - r^{m+1}}{1 - r}$$

Solving the Recursion Tree

2. $a/b^d < 1$

   Let $r = a/b^d$ and $m = log_b(n)$

   Use the closed form solution: $\sum_{i=0}^{m} r^i = \frac{r^{m+1}-1}{r-1}$

$$T(n) = n^d \cdot \sum_{i=0}^{m} r^i$$

$$= n^d \cdot \frac{r^{m+1} - 1}{r - 1} \cdot \frac{-1}{-1}$$

$$= n^d \cdot \frac{1 - r^{m+1}}{1 - r}$$

$$\frac{1 - r^{0+1}}{1 - r} \leq \frac{1 - r^{m+1}}{1 - r}$$

Solving the Recursion Tree

2. $a/b^d < 1$

   Let $r = a/b^d$ and $m = log_b(n)$

   Use the closed form solution: $\sum_{i=0}^{m} r^i = \frac{r^{m+1}-1}{r-1}$

$$
\begin{aligned}
T(n) &= n^d \cdot \sum_{i=0}^{m} r^i \\
&= n^d \cdot \frac{r^{m+1} - 1}{r - 1} \cdot \frac{-1}{-1} \\
&= n^d \cdot \frac{1 - r^{m+1}}{1 - r}
\end{aligned}
$$

$$1 = \frac{1 - r^{0+1}}{1 - r} \leq \frac{1 - r^{m+1}}{1 - r}$$

Solving the Recursion Tree

2. $a/b^d < 1$

   Let $r = a/b^d$ and $m = log_b(n)$

   Use the closed form solution: $\sum_{i=0}^{m} r^i = \frac{r^{m+1}-1}{r-1}$

$$T(n) = n^d \cdot \sum_{i=0}^{m} r^i$$

$$= n^d \cdot \frac{r^{m+1} - 1}{r - 1} \cdot \frac{-1}{-1}$$

$$= n^d \cdot \frac{1 - r^{m+1}}{1 - r}$$

$$1 = \frac{1 - r^{0+1}}{1 - r} \leq \frac{1 - r^{m+1}}{1 - r} < \frac{1 - 0}{1 - r}$$

Note: since $r < 1$, $r^{m+1} \ll 1$

Solving the Recursion Tree

2. $a/b^d < 1$

   Let $r = a/b^d$ and $m = log_b(n)$

   Use the closed form solution: $\sum_{i=0}^{m} r^i = \frac{r^{m+1}-1}{r-1}$

$$T(n) = n^d \cdot \sum_{i=0}^{m} r^i$$

$$= n^d \cdot \frac{r^{m+1} - 1}{r - 1} \cdot \frac{-1}{-1}$$

$$= n^d \cdot \frac{1 - r^{m+1}}{1 - r}$$

$$1 = \frac{1 - r^{0+1}}{1 - r} \leq \frac{1 - r^{m+1}}{1 - r} < \frac{1 - 0}{1 - r} = \frac{1}{1 - a/b^d}$$

Solving the Recursion Tree

2. $a/b^d < 1$

   Let $r = a/b^d$ and $m = log_b(n)$

   Use the closed form solution: $\sum_{i=0}^{m} r^i = \frac{r^{m+1}-1}{r-1}$

$$T(n) = n^d \cdot \sum_{i=0}^{m} r^i$$

$$= n^d \cdot \frac{r^{m+1} - 1}{r - 1} \cdot \frac{-1}{-1}$$

$$= n^d \cdot \frac{1 - r^{m+1}}{1 - r}$$

$$1 = \frac{1 - r^{0+1}}{1 - r} \leq \frac{1 - r^{m+1}}{1 - r} < \frac{1 - 0}{1 - r} = \frac{1}{1 - a/b^d}$$

Hence $T(n)$ is $\Theta\left(n^d\right)$

Solving the Recursion Tree

3. $a/b^d > 1$

   Let $r = a/b^d$ and $m = log_b(n)$

Solving the Recursion Tree

3. $a/b^d > 1$

   Let $r = a/b^d$ and $m = log_b(n)$

$$T(n) \quad = n^d \cdot \sum_{i=0}^{m} r^i$$

Solving the Recursion Tree

3. $a/b^d > 1$

   Let $r = a/b^d$ and $m = log_b(n)$

$$T(n) = n^d \cdot \sum_{i=0}^{m} r^i$$

$$= n^d \cdot \frac{r^{m+1} - 1}{r - 1}$$

Solving the Recursion Tree

3. $a/b^d > 1$

   Let $r = a/b^d$ and $m = log_b(n)$

$$
\begin{aligned}
T(n) &= n^d \cdot \sum_{i=0}^{m} r^i \\
&= n^d \cdot \frac{r^{m+1} - 1}{r - 1} \\
&= n^d \cdot \Theta(r^{m+1})
\end{aligned}
$$

Solving the Recursion Tree

3. $a/b^d > 1$

   Let $r = a/b^d$ and $m = log_b(n)$

$$
\begin{aligned}
T(n) &= n^d \cdot \sum_{i=0}^{m} r^i \\
&= n^d \cdot \frac{r^{m+1} - 1}{r - 1} \\
&= n^d \cdot \Theta(r^{m+1}) \\
&= n^d \cdot \Theta(r^m)
\end{aligned}
$$

Solving the Recursion Tree

3. $a/b^d > 1$
   Let $r = a/b^d$ and $m = log_b(n)$

$$
\begin{aligned}
T(n) &= n^d \cdot \sum_{i=0}^{m} r^i \\
&= n^d \cdot \frac{r^{m+1} - 1}{r - 1} \\
&= n^d \cdot \Theta(r^{m+1}) \\
&= n^d \cdot \Theta(r^m)
\end{aligned}
$$

$r^m \geq \frac{1}{r} \cdot r \cdot r^m$ when $m \geq 1$

Solving the Recursion Tree

3. $a/b^d > 1$
   Let $r = a/b^d$ and $m = log_b(n)$

$$T(n) = n^d \cdot \sum_{i=0}^{m} r^i$$
$$= n^d \cdot \frac{r^{m+1} - 1}{r - 1}$$
$$= n^d \cdot \Theta(r^{m+1})$$
$$= n^d \cdot \Theta(r^m)$$

$$r^m \geq \frac{1}{r} \cdot r \cdot r^m \quad \text{when } m \geq 1$$
$$\geq \frac{1}{r} \cdot r^{m+1} \quad \text{when } m \geq 1$$

Solving the Recursion Tree

3. $a/b^d > 1$
   Let $r = a/b^d$ and $m = log_b(n)$

$$
\begin{aligned}
T(n) &= n^d \cdot \sum_{i=0}^{m} r^i \\
&= n^d \cdot \frac{r^{m+1} - 1}{r - 1} \\
&= n^d \cdot \Theta(r^{m+1}) \\
&= n^d \cdot \Theta(r^m)
\end{aligned}
$$

r^m	$\geq \frac{1}{r} \cdot r \cdot r^m$	when $m \geq 1$
	$\geq \frac{1}{r} \cdot r^{m+1}$	when $m \geq 1$
r^m	is $\Omega(r^{m+1})$	for witnesses $c = \frac{1}{r}$ and $m_0 = 1$

Solving the Recursion Tree

3. $a/b^d > 1$

   Let $r = a/b^d$ and $m = log_b(n)$

$$
\begin{aligned}
T(n) &= n^d \cdot \sum_{i=0}^{m} r^i \\
&= n^d \cdot \frac{r^{m+1} - 1}{r - 1} \\
&= n^d \cdot \Theta(r^{m+1}) \\
&= n^d \cdot \Theta(r^m) \\
&= n^d \cdot \Theta\left(\left(a/b^d\right)^{log_b(n)}\right)
\end{aligned}
$$

$$
\begin{aligned}
r^m &\geq \frac{1}{r} \cdot r \cdot r^m && \text{when } m \geq 1 \\
&\geq \frac{1}{r} \cdot r^{m+1} && \text{when } m \geq 1
\end{aligned}
$$

r^m is $\Omega(r^{m+1})$ for witnesses $c = \frac{1}{r}$ and $m_0 = 1$

Solving the Recursion Tree

3. $a/b^d > 1$
   Let $r = a/b^d$ and $m = log_b(n)$

$$T(n) = n^d \cdot \sum_{i=0}^{m} r^i$$
$$= n^d \cdot \frac{r^{m+1} - 1}{r - 1}$$
$$= n^d \cdot \Theta(r^{m+1})$$
$$= n^d \cdot \Theta(r^m)$$
$$= n^d \cdot \Theta\left(\left(a/b^d\right)^{log_b(n)}\right)$$
$$= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{b^{log_b(n) \cdot d}}\right)$$

r^m	$\geq \frac{1}{r} \cdot r \cdot r^m$	when $m \geq 1$	
	$\geq \frac{1}{r} \cdot r^{m+1}$	when $m \geq 1$	
r^m	is $\Omega(r^{m+1})$	for witnesses $c = \frac{1}{r}$ and $m_0 = 1$	

Solving the Recursion Tree

3. $a/b^d > 1$

   Let $r = a/b^d$ and $m = log_b(n)$

$$T(n) \quad = n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{b^{log_b(n)\cdot d}}\right)$$

Solving the Recursion Tree

3. $a/b^d > 1$
   Let $r = a/b^d$ and $m = log_b(n)$

$$T(n) = n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{b^{log_b(n)\cdot d}}\right)$$

$$= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{\left(b^{log_b(n)}\right)^d}\right)$$

Solving the Recursion Tree

3. $a/b^d > 1$

   Let $r = a/b^d$ and $m = log_b(n)$

$$T(n) = n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{b^{log_b(n)\cdot d}}\right)$$

$$= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{\left(b^{log_b(n)}\right)^d}\right)$$

$$= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{n^d}\right)$$

Solving the Recursion Tree

3. $a/b^d > 1$
   Let $r = a/b^d$ and $m = log_b(n)$

$$T(n) = n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{b^{log_b(n) \cdot d}}\right)$$

$$= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{\left(b^{log_b(n)}\right)^d}\right)$$

$$= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{n^d}\right)$$

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

Solving the Recursion Tree

3. $a/b^d > 1$
   Let $r = a/b^d$ and $m = log_b(n)$

$$T(n) \quad = n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{b^{log_b(n)\cdot d}}\right)$$

$$= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{\left(b^{log_b(n)}\right)^d}\right)$$

$$= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{n^d}\right)$$

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

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

Solving the Recursion Tree

3. $a/b^d > 1$
   Let $r = a/b^d$ and $m = log_b(n)$

$$T(n) = n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{b^{log_b(n)\cdot d}}\right)$$

$$= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{\left(b^{log_b(n)}\right)^d}\right)$$

$$= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{n^d}\right)$$

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

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

$$log(x^y) = y \cdot log(x)$$

Solving the Recursion Tree

3. $a/b^d > 1$

   Let $r = a/b^d$ and $m = log_b(n)$

$$T(n) = n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{b^{log_b(n)\cdot d}}\right)$$

$$= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{\left(b^{log_b(n)}\right)^d}\right)$$

$$= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{n^d}\right)$$

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

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

$$log(x^y) = y \cdot log(x)$$

$$log_b(n) \cdot log_b(a) = log_b(a) \cdot log_b(n)$$

Solving the Recursion Tree

3. $a/b^d > 1$
   Let $r = a/b^d$ and $m = log_b(n)$

$$
\begin{aligned}
T(n) &= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{b^{log_b(n)\cdot d}}\right) \\
&= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{\left(b^{log_b(n)}\right)^d}\right) \\
&= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{n^d}\right) \\
&= \Theta\left(a^{log_b(n)}\right) \\
&= \Theta\left(n^{log_b(a)}\right)
\end{aligned}
$$

$$
\begin{aligned}
log(x^y) &= y \cdot log(x) \\
\\
log_b(n) \cdot log_b(a) &= log_b(a) \cdot log_b(n) \\
log_b\left(a^{log_b(n)}\right) &= log_b\left(n^{log_b(a)}\right)
\end{aligned}
$$

Solving the Recursion Tree

3. $a/b^d > 1$
   Let $r = a/b^d$ and $m = log_b(n)$

$$
\begin{aligned}
T(n) &= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{b^{log_b(n)\cdot d}}\right) \\
&= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{\left(b^{log_b(n)}\right)^d}\right) \\
&= n^d \cdot \Theta\left(\frac{a^{log_b(n)}}{n^d}\right) \\
&= \Theta\left(a^{log_b(n)}\right) \\
&= \Theta\left(n^{log_b(a)}\right)
\end{aligned}
$$

$$
\begin{aligned}
log(x^y) &= y \cdot log(x) \\
\\
log_b(n) \cdot log_b(a) &= log_b(a) \cdot log_b(n) \\
log_b\left(a^{log_b(n)}\right) &= log_b\left(n^{log_b(a)}\right) \\
a^{log_b(n)} &= n^{log_b(a)}
\end{aligned}
$$

The Master Theorem

Consider a recurrence relation and initial condition of the following form where a, b, and d are constants:

$$T(1) \text{ is a constant}$$
$$T(n) = aT(n/b) + \Theta\left(n^d\right)$$

1. If $a/b^d = 1$, then $T(n)$ is $\Theta\left(n^d log(n)\right)$
2. If $a/b^d < 1$, then $T(n)$ is $\Theta\left(n^d\right)$
3. If $a/b^d > 1$, then $T(n)$ is $\Theta\left(n^{log_b(a)}\right)$