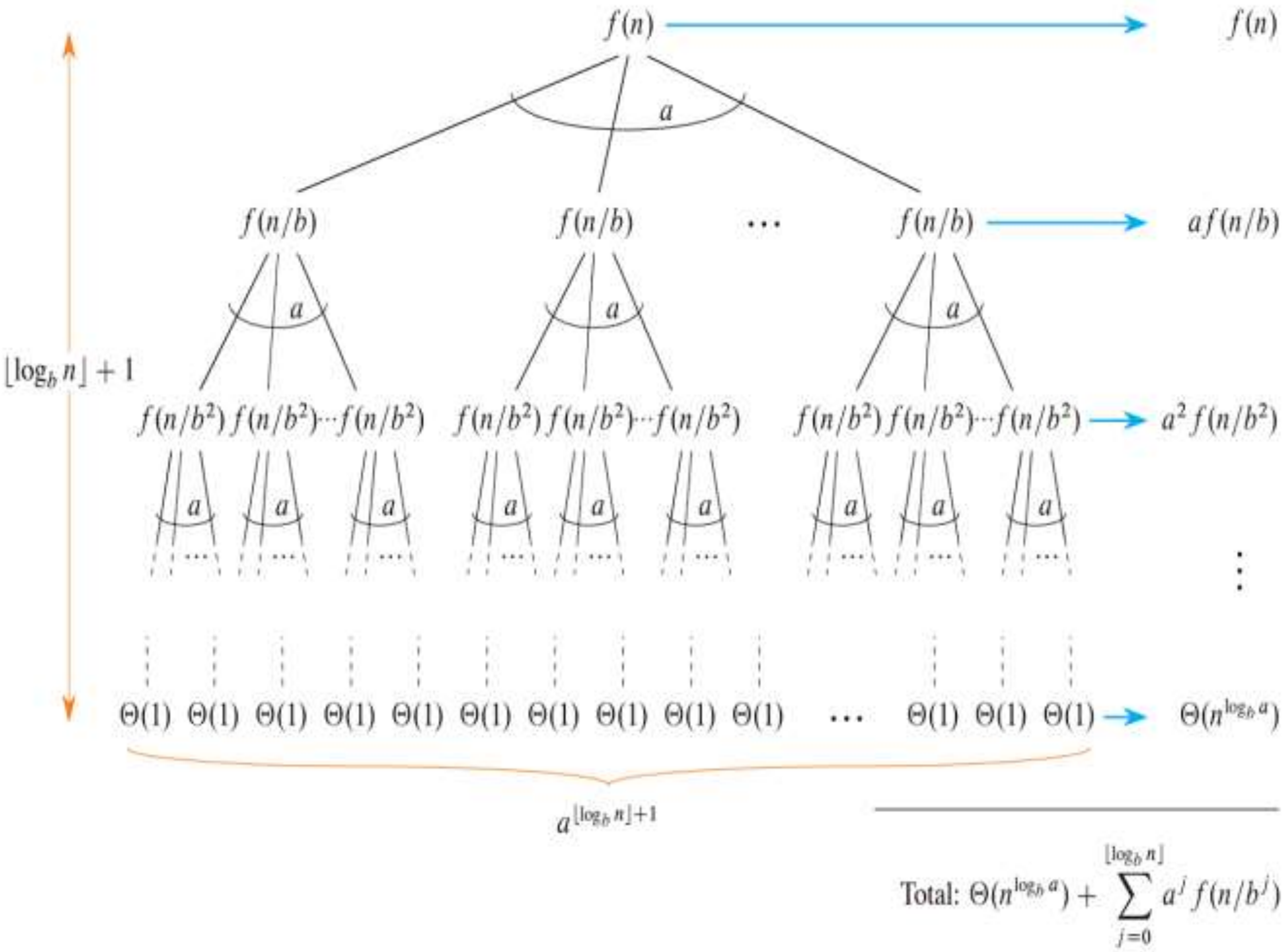
Divide And Conquer −
Continuous Master Theorem or
Generalized Master Theorem -Proof.

A generalized version of the Akra − Bazzi theorem has been given by Cormen in his book Introduction to Algorithms. The advantage of this modified master theorem is that it avoids the integration required in the Akra − Bazzi method.

The generalized master theorem proofis given as follows:

Let $a > 0$ ***and*** $b > 1$ ***be constants, and let numbers*** $n \geq 1$.
Then the recurrence:

$$T(n) = \begin{cases} \Theta(1) & \text{if } 0 \leq n < 1 \\ aT\left(\frac{n}{b}\right) + f(n) & \text{if } n \geq 1 \end{cases}$$



Proof:

Consider the recursion tree in the Figure. Let's look at its internal nodes. The root of the tree has cost $f(n)$, and it has a children, each with cost $f\left(\frac{n}{b}\right)$. (It is convenient to think of `a` as being an integer, especially when visualizing the recursion tree, but the mathematics doesnot require it.) Each of these children has a children, making a^2 nodes at depth 2, and each of the `a` children has cost $f\left(\frac{n}{b^2}\right)$.

$In\ general\ , there\ are\ a^j\ nodes\ at\ depth\ j\ , and\ each\ node\ has$
$cost\ f\left(\frac{n}{b^j}\right).$

$Now, let's\ move\ on\ to\ understanding\ the\ leaves\ . The\ tree\ grows$
$downward\ until\ \frac{n}{b^j}\ \ becomes\ less\ than\ 1.$

$Thus, the\ tree\ has\ height\ \ \lfloor \log_b n \rfloor + 1, because$
$\frac{n}{b^{\lfloor \log_b n \rfloor}} \geq \frac{n}{b^{\log_b n}} = 1\ and\ \frac{n}{b^{\lfloor \log_b n \rfloor + 1}} < \frac{n}{b^{\log_b n}} = 1. Since, as\ we$
$have\ observed, the\ number\ of\ nodes\ at\ depth\ `j`\ is\ `\ a^j\ `\ \ and\ all\ the$
$leaves\ are\ at\ depth\ \lfloor \log_b n \rfloor + 1, the\ tree\ contains\ a^{\lfloor \log_b n \rfloor + 1}$
$leaves. Using\ the\ identity, we\ have\ a^{\lfloor \log_b n \rfloor + 1} \leq a^{\log_b n} = an^{\log_b a}$
$i.e.,$

$[The\ logarithm\ \log_b n\ \ represents\ the\ exponent\ to\ which\ we\ raise$
$the\ \ `b`\ to\ obtain\ `n`\ . The\ floor\ function\ \ \lfloor \mathrm{x} \rfloor\ rounds\ down\ `x`\ to\ the$
$nearest\ integer. Therefore\ \lfloor \log_b n \rfloor \leq\ \ \log_b n\ \ . Hence$
$a^{\lfloor \log_b n \rfloor + 1} \leq a^{\log_b n}\ . We\ also\ know, by\ property\ of\ logarithm$
$b^{\log_b n} = n\ \ and\ observe\ that\ a^{\ \lfloor \log_b n \rfloor + 1}\ and\ a^{\log_b n}\ have\ same$
$base\ `a`. Since\ \ \lfloor \log_b n \rfloor\ is\ the\ largest\ integer \leq\ \log_b n,$
$adding\ 1\ to\ it\ will\ still\ be\ less\ than\ or\ equal\ to\ \ \log_b n.$
$We\ know\ that\ b^{\log_b n} = n\ and\ b^{\log_b a} = a\ , substituting$
$these\ values\ into\ the\ inequality, we\ have$
$a^{\lfloor \log_b n \rfloor + 1} \leq a^{\log_b n} \Longrightarrow\ a^{\lfloor \log_b n \rfloor + 1} \leq n^{\log_b a}. Since$

$\lfloor \log_b n \rfloor$ *is an integer and* $\log_b a$ *is a constant, we can rewrite*:

$\lfloor \log_b n \rfloor + 1 = \log_b n + 1$ *, hence by taking the base*

\`a\` logarithm of both sides, we obtain $\lfloor \log_b n \rfloor + 1 \leq$

$\log_b a \times \log_b n$ *. Combining these steps, we have*

$a^{\log_b n} = n^{\log_b a} \geq a^{\log_b n}$*, which demonstrates that*

$a^{\lfloor \log_b n \rfloor + 1} \leq a^{\log_b n}$] .

Now $a^{\lfloor \log_b n \rfloor + 1} \leq a^{\log_b n} = an^{\log_b a} = O\left(n^{\log_b a}\right)$*, since* a *is*

constant and $a^{\lfloor \log_b n \rfloor + 1} \geq a^{\log_b n} = n^{\log_b a} = \Omega\left(n^{\log_b a}\right)$.

Consequently the total number of leaves is $\Theta\left(n^{\log_b a}\right)$

−asymptotically, the watershed function.

We now in a position to derive equation:

<u>***Equation 1***</u>

Let $a > 0$ *and* $b > 1$ *be constants, and let* $f(n)$ *be a function*

defined over real numbers $n \geq 1$*. Then the recurrence*:

$$T(n) = \begin{cases} \Theta(1) & \text{if } 0 \leq n < 1 \\ aT\left(\frac{n}{b}\right) + f(n) & \text{if } n \geq 1 \end{cases}$$

$has\ solution$:

$$T(n) = \Theta\left(n^{\log_b a}\right) + \sum_{j=0}^{\lfloor \log_b n \rfloor} a^j f\left(\frac{n}{b^j}\right)$$

$Hence\ by\ summing\ the\ cost\ of\ the\ nodes\ at\ each\ depth\ in\ the\ tree\ as\ shown\ in\ the\ figure. The\ first\ term\ in\ the\ equation\ is\ the\ total\ cost\ of\ the\ leaves. Since\ each\ leaf\ is\ at\ depth\ \lfloor \log_b n \rfloor + 1\ and$

$\frac{n}{b^{\lfloor \log_b n \rfloor + 1}} < 1\ , the\ base\ case\ of\ the\ recurrence\ gives\ the\ cost\ of\ a\ leaf: T\left(\frac{n}{b^{\lfloor \log_b n \rfloor + 1}}\right) = \Theta(1). Hence\ the\ cost\ of\ all\ \Theta\left(\mathrm{n}^{\log_b a}\right)\ leaves\ is\ \Theta\left(\mathrm{n}^{\log_b a}\right) \times \Theta(1) = \Theta\left(\mathrm{n}^{\log_b a}\right).$

$Lets\ view\ the\ division\ again$:

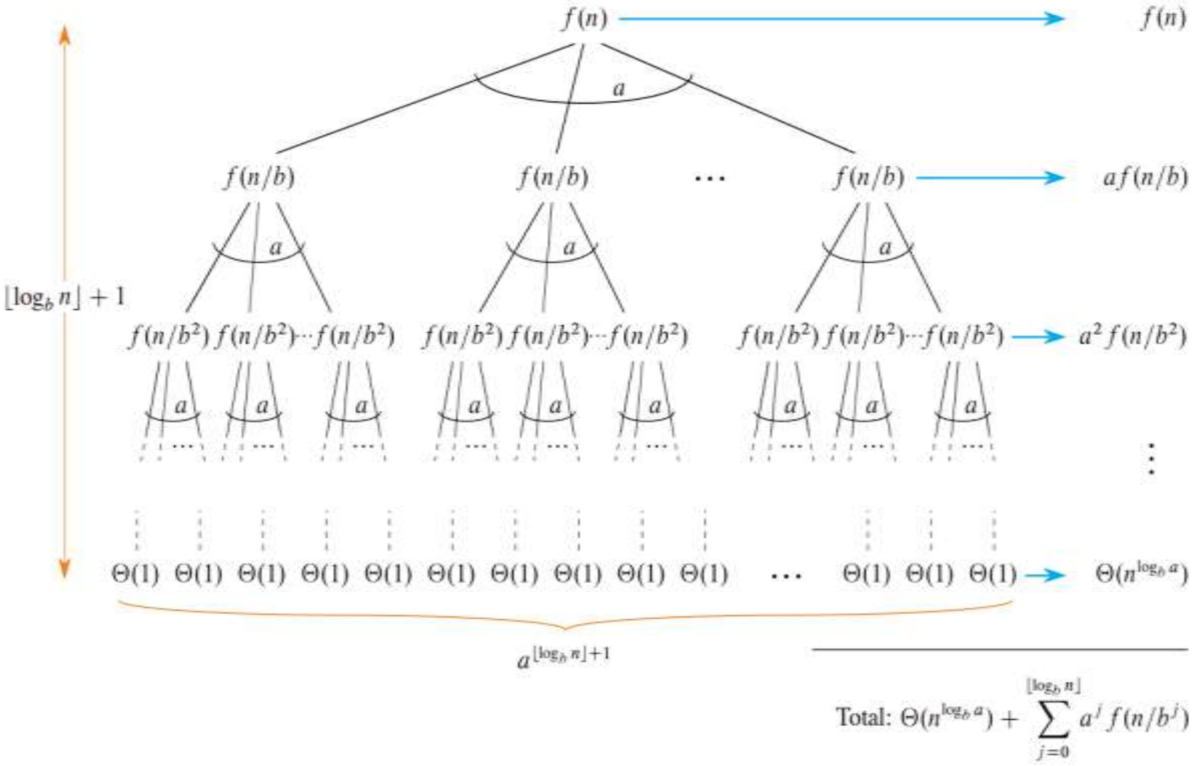


Figure: The recursion tree generated by $T(n) = aT(n/b) + f(n)$. The tree is a complete a-ary tree with $a^{\lfloor \log_b n \rfloor + 1}$ leaves and height $\lfloor \log_b n \rfloor + 1$.

Level	***No. of problems***	***Problem Size***	***Work done = Problem Size × No. of Problems***
$\mathbf{0}$	$\mathbf{1}$	$\boldsymbol{f(n)}$	$\mathbf{1} \times \boldsymbol{f(n)} = \boldsymbol{f(n)}$
$\mathbf{1}$	$\boldsymbol{a}$	$\boldsymbol{f\left(\frac{n}{b}\right)}$	$\boldsymbol{a \times f\left(\frac{n}{b}\right)}$ $\boldsymbol{= af\left(\frac{n}{b}\right)}$

$\mathbf{2}$	a^2	$f\left(\frac{n}{b^2}\right)$	$a^2 \times f\left(\frac{n}{b^2}\right) =$ $a^2 f\left(\frac{n}{b^2}\right)$
. .	. .	. .	. .
$n-1$	a^k	$f\left(\frac{n}{b^k}\right)$	$a^k \times f\left(\frac{n}{b^k}\right) =$ $a^k f\left(\frac{n}{b^k}\right)$
$k = \log_b n$	$a^{\log_b n}$	$f\left(\frac{n}{b^{\log_b n}}\right)$	$a^{\log_b n} \times f\left(\frac{n}{b^{\log_b n}}\right)$ $= a^{\log_b n} f\left(\frac{n}{b^{\log_b n}}\right)$

$That\ is\ we\ get:$

$$f(n) + a^2 f\left(\frac{n}{b^2}\right) + \cdots + a^{\log_b n} \times f\left(\frac{n}{b^{\log_b n}}\right)$$

$As\ we\ know\ no.of\ levels\ of\ no.of\ height\ of\ the\ tree$

$= \log_b n + 1\ \ or\ \lfloor \log_b n \rfloor + 1$

And the tree is a complete $a-$ary tree with $a^{\log_b n+1}$ or $a^{\lfloor \log_b n \rfloor +1}$.

Then recombining the subproblems . Since the cost for all the internal nodes at depth j as $a^j f\left(\frac{n}{b^j}\right)$, the total cost of all internal nodes is:

$$\sum_{j=0}^{\lfloor \log_b n \rfloor} a^j f\left(\frac{n}{b^j}\right)$$

As we will see, the three cases of the master theorem depend on the distribution of the total cost across levels of the recursion tree:

Case1: The cost increase geometrically from the root to the leaves, growing by a constant factor with each level.

Case 2: The costs depend on the value of \`k\` in the theorem. With $k = 0$, the costs are equal for each level ; with $k = 1$, the costs grow linearly from the root to the leaves; with $k = 2$; the growth is quadratic; and in general , the costs grow polynomially in k.

Case 3: The costs decrease geometrically from the root to the leaves, shrinking by a constant factor

$with\ each\ level.$

$The\ summation\ equation:$

$$T(n) = \Theta\left(n^{\log_b a}\right) + \sum_{j=0}^{\lfloor \log_b n \rfloor} a^j f\left(\frac{n}{b^j}\right)$$

$It\ describes\ the\ cost\ of\ the\ dividing\ and\ combiningsteps\ in\ the\ underlying\ divide\ and\ conquer\ algorithm.$

$\underline{PART\ 1:}$

$Let\ a > 0\ and\ b > 1\ be\ constants, and\ let\ f(n)\ be\ a\ function\ defined\ over\ real\ numbers\ n \geq 1. Then\ the\ asymptotic\ behavior\ of\ the\ function:$

$$g(n) = \sum_{j=0}^{\lfloor \log_b n \rfloor} a^j f\left(\frac{n}{b^j}\right),$$

$defined\ for\ n \geq 1, can\ be\ characterised\ as\ follows:$

$1.$ *If there exists a constant* $\varepsilon > 0$ *such that* $f(n) = O\left(n^{\log_b a - \varepsilon}\right)$, *then* $g(n) = O\left(n^{\log_b a}\right)$.

$2.$ *If there exists a constant* $k \geq 0$ *such that* $f(n) = O\left(n^{\log_b a} \log^k n\right)$, *then* $g(n) = O\left(n^{\log_b a} \log^{k+1} n\right)$.

$3.$ *If there exists a constant `c` in the range* $0 < c < 1$ *such that* $0 < af\left(\frac{n}{b}\right) \leq cf(n)$ *for all* $n \geq 1$, *then* $g(n) = \Theta(f(n))$.

Proof:

For case 1, *we have* $f(n) = O\left(n^{\log_b a - \varepsilon}\right)$, *which implies that* $f\left(\frac{n}{b^j}\right) = O\left(\left(\frac{n}{b^j}\right)^{\log_b a - \varepsilon}\right)$. *Substituting into equation*:

$$\sum_{j=0}^{\lfloor \log_b n \rfloor} a^j f\left(\frac{n}{b^j}\right)$$

, *which yields*:

$$g(n) = \sum_{j=0}^{\lfloor \log_b n \rfloor} a^j O\left(\left(\frac{n}{b^j}\right)^{\log_b a - \varepsilon}\right)$$

$$= \sum_{j=0}^{\lfloor \log_b n \rfloor} a^j \, O\left(\left(\frac{n}{b^j}\right)^{\log_b a - \varepsilon}\right)$$

$$= O\left(\sum_{j=0}^{\lfloor \log_b n \rfloor} a^j \left(\frac{n}{b^j}\right)^{\log_b a - \varepsilon}\right)$$

$$= O\left(\sum_{j=0}^{\lfloor \log_b n \rfloor} a^j \frac{n^{\log_b a - \varepsilon}}{(b^j)^{\log_b a - \varepsilon}}\right)$$

$$= O\left(n^{\log_b a - \varepsilon} \sum_{j=0}^{\lfloor \log_b n \rfloor} a^j \frac{1}{(b^j)^{\log_b a - \varepsilon}}\right)$$

$$= O\left(n^{\log_b a - \varepsilon} \sum_{j=0}^{\lfloor \log_b n \rfloor} \frac{a^j}{(b^j)^{\log_b a - \varepsilon}}\right)$$

$$= O\left(n^{\log_b a - \varepsilon} \sum_{j=0}^{\lfloor \log_b n \rfloor} \left(\frac{a}{b^{\log_b a - \varepsilon}}\right)^j\right)$$

$we\ know:\ b^{\log_b a} \times \frac{1}{b^{\varepsilon}} = b^{\log_b a - \varepsilon}. Then,$

$$= O\left(n^{\log_b a-\varepsilon}\sum_{j=0}^{\lfloor\log_b n\rfloor}\left(\frac{a}{b^{\log_b a}\times\frac{1}{b^{\varepsilon}}}\right)^j\right)$$

$$= O\left(n^{\log_b a-\varepsilon}\sum_{j=0}^{\lfloor\log_b n\rfloor}\left(\frac{ab^{\varepsilon}}{b^{\log_b a}}\right)^j\right)$$

$We\ know:\ b^{\log_b a} = a\ then:$

$$= O\left(n^{\log_b a-\varepsilon}\sum_{j=0}^{\lfloor\log_b n\rfloor}\left(\frac{ab^{\varepsilon}}{a}\right)^j\right)$$

$$= O\left(n^{\log_b a-\varepsilon}\sum_{j=0}^{\lfloor\log_b n\rfloor}(b^{\varepsilon})^j\right)$$

$$\Rightarrow \sum_{j=0}^{\lfloor\log_b n\rfloor}(b^{\varepsilon})^j = (b^{\varepsilon})^0 + (b^{\varepsilon})^1 + \cdots + (b^{\varepsilon})^{\lfloor\log_b n\rfloor}$$

$$= 1 + (b^{\varepsilon})^1 + (b^{\varepsilon})^2 + \cdots + (b^{\varepsilon})^{\lfloor\log_b n\rfloor}$$

$Hence\ the\ above\ series\ is\ geometric\ expression:$

$And\ we\ know:$

$$\sum_{k=0}^{n} x^k = 1 + x + x^2 + \cdots + x^n = \frac{x^{n+1} - 1}{x - 1}$$

$\boldsymbol{Hence}$:

$$\Rightarrow O\left(n^{\log_b a-\varepsilon}\left(\frac{b^{\varepsilon(\lfloor\log_b n\rfloor+1)} - 1}{b^{\varepsilon} - 1}\right)\right)$$

As the series is geometric . Since `b` and `ε` are constants ,
the $b^{\varepsilon} - 1$ *dominator doesnot affect the asymptotic growth*
of $g(n)$ *and neither does the* -1 *in the numerator*.

Since $b^{\varepsilon(\lfloor\log_b n\rfloor+1)} \le \left(b^{\log_b n+1}\right)^{\varepsilon}$

$$\Rightarrow \left(b^{\log_b n}\right)^{\varepsilon} \times b^{\varepsilon}$$

$$\Rightarrow (n)^{\varepsilon} \times b^{\varepsilon}\left[As\ b^{\log_b n} = n\right]$$
$$\Rightarrow O(n^{\varepsilon} b^{\varepsilon})$$
$$\Rightarrow O(n^{\varepsilon})\ [As\ b\ is\ constant\ here]$$

$\boldsymbol{Hence\ we\ obtain}$:

$$g(n) = O\left(n^{\log_b a-\varepsilon} \times O(n^{\varepsilon})\right)$$

$$\Longrightarrow O\left(n^{\log_b a-\varepsilon+\varepsilon}\right)$$

$$\Longrightarrow O\left(n^{\log_b a}\right)$$

$$\textit{Thereby proving case } 1.$$

Case 2 assumes that $f(n) = \Theta\left(n^{\log_b a}\, log^k n\right)$*, from which we can conclude that* $f\left(\frac{n}{b^j}\right) = \Theta\left(\left(\frac{n}{b^j}\right)^{\log_b a}\ log^k\ \left(\frac{n}{b^j}\right)\right)$. *i.e. substituting*:

$$g(n) = \sum_{j=0}^{\lfloor\log_b n\rfloor} a^j\, f\left(\frac{n}{b^j}\right)$$

Hence substituting the above equation again and again it yields:

$$g(n) = \Theta\left(\sum_{j=0}^{\lfloor\log_b n\rfloor} a^j\ \left(\frac{n}{b^j}\right)^{\log_b a}\ log^k\ \left(\frac{n}{b^j}\right)\right)$$

$$= \Theta\left(\sum_{j=0}^{\lfloor \log_b n \rfloor} a^j \left(\frac{n^{\log_b a}}{(b^j)^{\log_b a}}\right) log^k \left(\frac{n}{b^j}\right)\right)$$

$$= \Theta\left(n^{\log_b a} \sum_{j=0}^{\lfloor \log_b n \rfloor} a^j \left(\frac{1}{(b^j)^{\log_b a}}\right) log^k \left(\frac{n}{b^j}\right)\right)$$

$$= \Theta\left(n^{\log_b a} \sum_{j=0}^{\lfloor \log_b n \rfloor} \left(\frac{a^j}{(b^j)^{\log_b a}}\right) log^k \left(\frac{n}{b^j}\right)\right)$$

$$= \Theta\left(n^{\log_b a} \sum_{j=0}^{\lfloor \log_b n \rfloor} \left(\frac{a}{b^{\log_b a}}\right)^j log^k \left(\frac{n}{b^j}\right)\right)$$

$$= \Theta\left(n^{\log_b a} \sum_{j=0}^{\lfloor \log_b n \rfloor} \left(\frac{a}{a}\right)^j log^k \left(\frac{n}{b^j}\right)\right) \left[b^{\log_b a} = a\right]$$

$$= \Theta\left(n^{\log_b a} \sum_{j=0}^{\lfloor \log_b n \rfloor} log^k \left(\frac{n}{b^j}\right)\right)$$

$$\boldsymbol{By\ applying}\ , \log_b a = \frac{\log_c a}{\log_c b}$$

$$= \Theta\left(n^{\log_b a} \sum_{j=0}^{\lfloor \log_b n \rfloor} \left(\frac{\log_b\left(\frac{n}{b^j}\right)}{\log_b 2}\right)^k\right) [As, \log_b 2 = 1]$$

$$= \Theta\left(n^{\log_b a} \sum_{j=0}^{\lfloor \log_b n \rfloor} \left(\frac{\log_b n - \log_b b^j}{\log_b 2}\right)^k\right) [Applying\ logarithmic\ rule:$$

$$\log_c\left(\frac{a}{b}\right) = \log_c a - \log_c b\ , where\ c\ is\ the\ base.]$$

$$= \Theta\left(n^{\log_b a} \sum_{j=0}^{\lfloor \log_b n \rfloor} \left(\frac{\log_b n - j}{\log_b 2}\right)^k\right)$$

$$= \Theta\left(n^{log_b a} \times \frac{1}{\log_b^k 2} \sum_{j=0}^{\lfloor log_b n \rfloor} (log_b n - j)^k\right)$$

$$= \Theta\left(n^{log_b a} \sum_{j=0}^{\lfloor log_b n \rfloor} (log_b n - j)^k\right) [b > 1\ and\ k\ are\ constants]$$

$The\ summation\ within\ the\ \Theta - notation\ can\ be\ bounded$
$above\ as\ follows$:

$$\sum_{j=0}^{\lfloor log_b n \rfloor} (log_b n - j)^k \leq \sum_{j=0}^{\lfloor log_b n \rfloor} (\lfloor \log_b n \rfloor + 1 - j)^k$$

$i.e. if\ we\ re - index\ we\ get(i.e. elemination\ of\ extra$
$(\lfloor \log_b n \rfloor + 1\ )\ from\ the\ series$:

$$\Rightarrow \sum_{j=1}^{\lfloor log_b n \rfloor + 1} (j)^k$$

$$\Rightarrow \sum_{j=1}^{\lfloor log_b n \rfloor + 1} (j)^k = 1^k + 2^k + \cdots + (\lfloor log_b n \rfloor + 1)^k$$

$i.e. the\ series\ was$: $\lfloor \log_b n \rfloor + 1 - \left(0^k + 1^k + \cdots + \ \lfloor \log_b n \rfloor^k\right)\ as$
$j = 0\ to\ \lfloor \log_b n \rfloor$

$Hence\ it\ is\ re - indexed\ j\ \ from\ 1\ to\ \lfloor log_b n \rfloor + 1\ now\ it\ is$:
$= \ 1^k + 2^k + \cdots + (\lfloor log_b n \rfloor + 1)^k$

$Now,$

$$\sum_{j=1}^{\lfloor log_b n \rfloor +1} (j)^k \leq \sum_{j=1}^{\lfloor log_b n \rfloor +1} (j)^{k+1}$$

$As\ the\ summation\ rapidly\ increases\ as\ j\ increases, hence\ the\ sum\ is\ dominated\ by\ largest\ terms.$

$The\ largest\ term\ for\ which\ j\ is\ colse\ to\ \ is : \lfloor log_b n \rfloor + 1\ , so\ the\ sum\ is : \mathrm{O}(\lfloor log_b n \rfloor + 1\ )^{k+1}$

$$= \mathrm{O}(log_b^{k+1} n)$$

$Similarly\ we\ can\ using\ the\ same\ above\ concept\ to\ show\ that\ \Omega\left(log_b^{k+1} n\right).$

$i.e.$

$$\sum_{j=0}^{\lfloor log_b n \rfloor} (\lfloor \log_b n \rfloor + 1 - j)^k \geq \sum_{j=0}^{\lfloor log_b n \rfloor} (log_b n - j)^k$$

$$\sum_{j=0}^{\lfloor log_b n \rfloor} (\lfloor \log_b n \rfloor + 1 - j)^k\ reindexed\ to : \sum_{j=1}^{\lfloor log_b n \rfloor +1} (j)^k$$

$Now,$

$$\sum_{j=1}^{\lfloor log_b n \rfloor + 1} (j)^{k+1} \geq \sum_{j=1}^{\lfloor log_b n \rfloor + 1} (j)^{k}$$

$$\Longrightarrow \Omega(log_b n)^{k+1}$$

$$\Longrightarrow \Omega(log_b^{k+1} n)$$

Since we have tight upper bound and tight lower bound the summation is: $\Theta(log_b^{k+1} n)$

From which we can conclude that :

$$\Theta\left(n^{log_b a} \sum_{j=0}^{\lfloor log_b n \rfloor} (log_b n - j)^k \right) =$$

$$\Longrightarrow \Theta\left(n^{log_b a} log_b^{k+1} n\right)$$

Thereby completing the proof of case 2.

$For\ case\ 3, observe\ that\ f(n)\ appears\ in\ definition:$

$$T(n) = \Theta\left(n^{\log_b a}\right) + \sum_{j=0}^{\lfloor \log_b n \rfloor} a^j f\left(\frac{n}{b^j}\right)$$

$$where\ , g(n) = \sum_{j=0}^{\lfloor \log_b n \rfloor} a^j f\left(\frac{n}{b^j}\right)$$

$(when\ j = 0,)\ and\ that\ all\ terms\ of\ g(n)\ are\ positive\ .$
$Therefore, we\ must\ have\ g(n) = \Omega\big(f(n)\big), and\ it\ only\ remains$
$to\ prove\ that\ g(n) = O\big(f(n)\big). Performing\ `j`\ iterations\ of\ the$
$inequality\ \ af\left(\frac{n}{b}\right) \le cf(n)\ yields\ a^j f\left(\frac{n}{b^j}\right) \le c^j f(n).$
$Substituting\ into\ the\ equation:$

$$g(n) = \sum_{j=0}^{\lfloor \log_b n \rfloor} a^j f\left(\frac{n}{b^j}\right)$$

$$\le \sum_{j=0}^{\lfloor \log_b n \rfloor} c^j f(n)$$

$$\leq f(n) \sum_{j=0}^{\infty} c^j$$

$\textit{Hence the series is}: 1 + c + c^2 + \cdots + \infty$.

The infinite decreasing geometric series occurs when the summation is infinite and $|x| < 1$:

$$\sum_{k=0}^{\infty} x^k = \frac{1}{1-x}$$

Hence applying it on above we get :

$$f(n) \sum_{j=0}^{\infty} c^j = f(n) \times \frac{1}{1-c}$$

$$\Rightarrow O(f(n)) \left[As \frac{1}{1-c} \text{ is constant} \right]$$

Thus we can conclude that $g(n) = \Theta\big(f(n)\big)$.

With this case 3 proved, the entire proof of PART 1 is complete.

Theorem (Continuous master theorem)

Let $a > 0$ and $b > 1$ be constants, and let $f(n)$ be a driving function that is defined and non-negative on all sufficiently large reals. Define the algorithmic recurrence $T(n)$ on the positive real numbers by:

$$T(n) = aT\left(\frac{n}{b}\right) + f(n)\,.$$

Then the asymptotic behavior of $T(n)$ can be characterized as follows:

1. If there exists a constant $\varepsilon > 0$ such that $f(n) = O\left(n^{\log_b a - \varepsilon}\right)$, then $T(n) = \Theta\left(n^{\log_b a}\right)$.

2. If there exists a constant $k \geq 0$ such that $f(n) = \Theta\left(n^{\log_b a} \log^k n\right)$, then $T(n) = \Theta\left(n^{\log_b a} \log^{k+1} n\right)$.

3. If there exists a constant $\varepsilon > 0$ such that $f(n) = \Omega\left(n^{\log_b a + \varepsilon}\right)$, and if $f(n)$ additionally satisfies the regularity condition $af\left(\frac{n}{b}\right) \leq cf(n)$ for some constant $c < 1$ and all sufficiently large n, then $T(n) = \Theta\left(f(n)\right)$.

$Proof$:

$The\ idea\ is\ to\ bound\ the\ summation$:

$$T(n) = \Theta\left(n^{\log_b a}\right) + \sum_{j=0}^{\lfloor \log_b n \rfloor} a^j f\left(\frac{n}{b^j}\right)$$

$and\ applying\ \ PART\ 1\ .But\ we\ must\ account\ the$
$above\ summation\ using\ a\ base\ case\ for\ 0 < n < 1,$
$whereas\ this\ theorem\ uses\ an\ implicit\ base\ case$
$for\ \ 0 < n < n_0\ , where\ n_0 > 0\ is\ an\ arbitrary$
$threshold\ constant. Since\ the\ recurrence\ is\ algorithmic\ ,$
$we\ can\ assume\ that\ f(n) is\ defined\ for\ n \geq n_0.$

$For\ n > 0\ , let\ us\ define\ two\ auxiliary\ T'(n) = T(n_0 n)$
$and\ f'(n) = f(n_0 n). We\ have$:

$$f'\left(\frac{n}{n_0}\right) = f(n)\ .$$

$$T'\left(\frac{n}{n_0}\right) = T(n)$$

$$T'(n) = T(n_0 n)$$

$$= \begin{cases} \Theta(1) & \text{if } n_0 n < n_0, \\ aT\left(\frac{n_0 n}{b}\right) + f(n_0 n) & \text{if } n_0 n \geq n_0, \end{cases}$$

$$= \begin{cases} \Theta(1) & \text{if } n < 1, \\ aT'\left(\frac{n}{b}\right) + f'(n) & \text{if } n \geq 1, \end{cases}$$

We have obtained a recurrence for $T'(n)$ that satisfies the condition, the solution gives:

$$T'(n) = \Theta\left(n^{\log_b a}\right) + \sum_{j=0}^{\lfloor \log_b n \rfloor} a^j f'\left(\frac{n}{b^j}\right)$$

To solve $T'(n)$, we first need to bound $f'(n)$. Lets examine the individual cases in the theorem.

$\underline{Case\ 1}$:

$The\ condition\ for\ case\ 1\ is\ f(n) = O\left(n^{log_b a-\varepsilon}\right)\ for$
$some\ constant\ \ \varepsilon > 0. We\ have$:

$$f'(n) = f(n_0 n)$$
$$= O\left((n_0 n)^{log_b a-\varepsilon}\right)$$
$$= O\left(n^{log_b a-\varepsilon}\right)$$

$since\ a, b, n_0\ and\ \varepsilon\ are\ all\ constant. The\ function\ f'(n)\ satisfies$
$the\ conditions\ of\ case\ 1\ of\ PART\ 1\ and\ the\ summation$:

$$T(n) = \Theta\left(n^{\log_b a}\right) + \sum_{j=0}^{\lfloor \log_b n \rfloor} a^j f\left(\frac{n}{b^j}\right)$$

$evaluates\ to\ O\left(n^{log_b a}\right). Because\ a, b, and\ n_0\ are\ all$
$constants, we\ have$:

$$T(n) = T'\left(\frac{n}{n_0}\right)$$
$$= \Theta\left(\left(\frac{n}{n_0}\right)^{log_b a}\right) + O\left(\left(\frac{n}{n_0}\right)^{log_b a}\right)$$

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

$= \Theta\left(\mathrm{n}^{log_b a}\right)\left[\begin{array}{c} same\ growth\ rate\ and\ \Theta\ \text{is present and preffered} \\ \text{before } O\ and\ \Theta \text{ represent both tight upper and tight} \\ lower\ bound\ while\ O\ represents\ only \text{ tight upper} \\ bound. \end{array}\right]$

thereby completing case 1 *of theorem*.

<u>*Case*2:</u>

The condition for case 2 *is* $f(n) = \Theta\left(n^{log_b a} log^k n\right)$ *for some constant* $k \geq 0$.

We have:

$f'(n) = f(n_0 n)$

$= \Theta\left((n_0 n)^{log_b a} log^k (n_0 n)\right)$

$= \Theta\left(\mathrm{n}^{log_b a}\, log^k n\right)$[*by eliminating the constant terms*].

Similar to the proof case 1 , *the function* $f'(n)$ *satisfies the conditions of case* 2 *of PART* 1. *The summation equation*:

$$T(n) = \Theta(n^{\log_b a}) + \sum_{j=0}^{\lfloor \log_b n \rfloor} a^j f\left(\frac{n}{b^j}\right)$$

is therefore $\Theta(n^{\log_b a} \log^{k+1} n)$, *which implies that* :

$$T(n) = T'\left(\frac{n}{n_0}\right)$$

$$= \Theta\left(\left(\frac{n}{n_0}\right)^{\log_b a}\right) + \Theta\left(\left(\frac{n}{n_0}\right)^{\log_b a} \log^{k+1}\left(\frac{n}{n_0}\right)\right)$$

$$= \Theta\left(\frac{1}{n_0} \times n^{\log_b a}\right) + \Theta\left(\frac{1}{n_0}\left(n^{\log_b a} \log^{k+1} n\right)\right)$$

$$= \Theta(n^{\log_b a}) + \Theta(n^{\log_b a} \log^{k+1} n)\ [\textit{eliminating constants}]$$

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

*Which proves case*2 *of the Theorem.*

<u>*Case* 3:</u>

Finally , the condition for case 3 *is* $f(n) = \Omega(n^{\log_b a + \varepsilon})$ *for some constant* $\varepsilon > 0$ *and* $f(n)$ *additionally satisfies the*

$regularity\ condition\ af\left(\frac{n}{b}\right) \leq cf(n)\ for\ all\ n \geq n_0\ and$

$some\ constant\ c < 1\ and\ n_0 > 1\ .The\ first\ part\ of\ case\ 3$

$is\ like\ case\ 1$:

$f'(n) = f(n_0 n)$

$= \ \Omega\left((n_0 n)^{log_b a+\varepsilon}\right)$

$= \ \Omega\left(n^{log_b a+\varepsilon}\right)\ [eliminating\ the\ constant\ n_0].$

$Using\ the\ definition\ f'(n)\ and\ the\ fact\ that\ n_0 n \geq n_0$

$for\ all\ n \geq 1, we\ have\ for\ n > 1\ that$:

$af'\left(\frac{n}{b}\right) = af\left(\frac{n_0 n}{b}\right)$

$\leq cf(n_0 n)$

$= cf'(n).$

$Thus\ f'(n)\ satisfies\ the\ requirements\ for\ case\ 3\ of\ Part\ 1$

$and\ the\ summation$:

$$T(n) = \ \Theta\left(n^{\log_b a}\right) + \sum_{j=0}^{\lfloor \log_b n \rfloor} a^j\, f\left(\frac{n}{b^j}\right)$$

$evaluates\ to\ \Theta\left(\mathrm{f}'(\mathrm{n})\right), yielding$

$$T(n) = T'\left(\frac{n}{n_0}\right)$$

$$= \Theta\left(\left(\frac{n}{n_0}\right)^{\log_b a}\right) + \Theta\left(f'\left(\frac{n}{n_0}\right)\right)$$

$$= \Theta\left(\left(\frac{n}{n_0}\right)^{\log_b a}\right) + \Theta\left(f'\left(\frac{n}{n_0}\right)\right)$$

$$= \Theta\left(\frac{1}{n_0}(n)^{\log_b a}\right) + \Theta\left(f'\left(\frac{n}{n_0}\right)\right)$$

$$= \Theta\left((n)^{\log_b a}\right) + \Theta\left(f'\left(\frac{n}{n_0}\right)\right) [\textit{Eliminating the constant}]$$

$\textit{We can write this as}$:

$$= \Theta\left(f'\left(\frac{n}{n_0}\right)\right) + \Theta\left((n)^{\log_b a}\right)$$

$\textit{As}$ $\Theta\left(f'\left(\frac{n}{n_0}\right)\right)$ $\textit{is assymptotically larger than}$ $\Theta\left((n)^{\log_b a}\right)$

$\textit{we will have}$:

$$= \Theta\left(f'\left(\frac{n}{n_0}\right)\right)$$

$$= \boldsymbol{\Theta}\big(f(n)\big)\ \left[As\ we\ know\ f(n) = \mathbf{f}'\left(\frac{\mathbf{n}}{\mathbf{n_0}}\right)\right].$$

$Which\ completes\ the\ proof\ of\ case\ 3\ of\ the\ theorem\ , and\ thus\ completes\ \ the\ proof\ of\ whole\ theorem.$

****