Example 4

Given recurrence relation is,

$$T(n) = 2T(\sqrt{n}) + C$$

Solve this eq^n using master's method.

First check if the equation is ~~not~~ in given format or not.

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

where $a \geq 1$, $b > 1$ & $f(n)$ = +ve fun^n

As the given equation ~~is in the format~~ we have

$a = 2$, $b = ?$

To get the value of b we have to make 2 assumptions,

Assume $\frac{n}{2^k} = 1$

$$\boxed{n = 2^k}$$

Taking log on the both sides

$$\log n = \log_2 2^k$$

$$\log n = k$$

$$\boxed{k = \log n}$$

The given relation is

$$T(n) = 2T(\sqrt{n}) + c$$

replace n with 2^k

$$T(2^k) = 2T(2^k)^{1/2} + c$$

$$T(2^k) = 2T(2^{k/2}) + c$$

Still the equation is not in the format required for solving it using master's method.

We dont have value for b

Now, Assume

$$\boxed{T(2^k) = S(k)}$$

$$T(2^k) = 2T(2^{k/2}) + c$$

Replace $T(2^k)$ with $S(k)$

$$\boxed{S(k) = 2\,S\left(\frac{k}{2}\right) + c}$$

Now the equation is in $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ format

So we have $a = 2$, $b = 2$, $f(n) = c$

$n^{\log_b a}$ we have $k^{\log_b a}$

$k^{\log_b a} = k^{\log_2 2}$

$= k^1$

$\boxed{k^{\log_b a} = k}$ —— This is $g(n)$

k is the value for n i.e. our $g(n)$

compare k & c, c is a constant with any values like 1, 2, 3

$c = 1$, k is greater than c as $k = \log n$

As per Case 3, we can say that

$\boxed{S(k) = k}$

But we need answer in terms of n

$S(k) = k$

$T(2^k) = k$

$\boxed{T(n) = \log n}$

So the time complexity is $\Theta(\log n)$.

$\boxed{T(n) = \Theta(\log n)}$

Example 5

$$T(n) = 2T(\sqrt{n}) + \log \cdot n$$

Solve it using master's method

As the given equation is not in the format $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$

To find value of b we have to make 2 assumptions.

Assume $\frac{n}{2^k} = 1$

$n = 2^k$

Taking log both sides we get

$\log n = k \cdot \log_2 2$

$\boxed{k = \log n}$ As $\log_2 2 = 1$

Now substitute $n = 2^k$ in the given equation

$$T(2^k) = 2T((2^k)^{1/2}) + \log 2^k$$

$$\boxed{T(2^k) = 2T(2^{k/2}) + \log 2^k}$$

~~This equation is~~ not ~~$aT(n/b) + f(n)$~~

~~Th~~

As this equation is not in form of
$T(n) = a \cdot T(n/b) + f(n)$

So let's make 2^{nd} assumption.

$$T(2^k) = S(k)$$

$$\boxed{S(k) = 2S(k/2) + \log_2 2^k}$$

Replacing $T(2^k)$ with $S(k)$
As, $\log_2 2 = 1$
Now the equation ~~is~~ becomes

$$\boxed{S(k) = 2S(k/2) + k}$$

Now this equation is in the form of
$T(n) = a \cdot T(n/b) + f(n)$

So let's solve it using master's theorem

$a = 2, \quad b = 2, \quad f(k) = k$

$n^{\log_b a}$

Here it is $k^{\log_b a}$ ~~=~~ ~~log~~

$= k^{\log_2 2} = k^1$

As $\log_2 2 = 1$

$\boxed{k^{\log_b a} = k}$ ... This is our $g(k)$

Let's compare $f(k)$ & $g(k)$

$f(k) = k$ & $g(k) = k$

So, $\boxed{f(k) = g(k)}$

So as per case-III of master's theorem we multiply the term with $\log n$

$$\boxed{S(k) = \Theta(k * \log k)}$$

But the given equation is in the form of $T(n)$

$$S(k) = \Theta(k * \log k)$$

$$T(2^k) = \Theta(\log n * \log \cdot \log n)$$

As, $S(k) = T(2^k)$ & $k = \log n$

Now we know $n = 2^k$

So, $\boxed{T(n) = \Theta(\log n \cdot \log \cdot \log n)}$