Master Theorem | Master Theorem Examples

Design & Analysis of Algorithms

Master Theorem-

Master’s theorem solves recurrence relations of the form-

$$T(n) = a\,T\left(\frac{n}{b}\right) + \theta\left(n^k \log^p n\right)$$

Master's Theorem

Here, a >= 1, b > 1, k >= 0 and p is a real number.

Master Theorem Cases-

To solve recurrence relations using Master’s theorem, we compare a with b^k.

Then, we follow the following cases-

Case-01:

If $a > b^k$, then $T(n) = \theta\ (n^{\log_b a})$

Case-02:

If $a = b^k$ and

- If $p < -1$, then $T(n) = \theta\ (n^{\log_b a})$
- If $p = -1$, then $T(n) = \theta\ (n^{\log_b a}.\log^2 n)$
- If $p > -1$, then $T(n) = \theta\ (n^{\log_b a}.\log^{p+1} n)$

Case-03:

If $a < b^k$ and

- If $p < 0$, then $T(n) = O\ (n^k)$
- If $p >= 0$, then $T(n) = \theta\ (n^k \log^p n)$

PRACTICE PROBLEMS BASED ON MASTER THEOREM-

Problem-01:

Solve the following recurrence relation using Master's theorem-

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

Solution-

We compare the given recurrence relation with $T(n) = aT(n/b) + \theta\ (n^k \log^p n)$.

Then, we have-

$$a = 3$$

$$b = 2$$

$$k = 2$$

$$p = 0$$

Now, $a = 3$ and $b^k = 2^2 = 4$.

Clearly, $a < b^k$.

So, we follow case-03.

Since $p = 0$, so we have-

$$T(n) = \theta\ (n^k \log^p n)$$

$$T(n) = \theta\ (n^2 \log^0 n)$$

Thus,

$$\mathbf{T(n) = \theta\ (n^2)}$$

Problem-02:

Solve the following recurrence relation using Master's theorem-

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

Solution-

We compare the given recurrence relation with $T(n) = aT(n/b) + \theta\ (n^k \log^p n)$.

Then, we have-

$$a = 2$$

$$b = 2$$

$$k = 1$$

$$p = 1$$

Now, $a = 2$ and $b^k = 2^1 = 2$.

Clearly, $a = b^k$.

So, we follow case-02.

Since $p = 1$, so we have-

$$T(n) = \theta\ (n^{\log_b a}.\log^{p+1} n)$$

$$T(n) = \theta\ (n^{\log_2 2}.\log^{1+1} n)$$

Thus,

$$\mathbf{T(n) = \theta\ (n\log^2 n)}$$

Problem-03:

Solve the following recurrence relation using Master's theorem-

$$T(n) = 2T(n/4) + n^{0.51}$$

Solution-

We compare the given recurrence relation with $T(n) = aT(n/b) + \theta(n^k \log^p n)$.

Then, we have-

$$a = 2$$

$$b = 4$$

$$k = 0.51$$

$$p = 0$$

Now, $a = 2$ and $b^k = 4^{0.51} = 2.0279$.

Clearly, $a < b^k$.

So, we follow case-03.

Since $p = 0$, so we have-

$$T(n) = \theta(n^k \log^p n)$$

$$T(n) = \theta(n^{0.51} \log^0 n)$$

Thus,

$$\boxed{\mathbf{T(n) = \theta(n^{0.51})}}$$

Problem-04:

Solve the following recurrence relation using Master's theorem-

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

Solution-

We compare the given recurrence relation with $T(n) = aT(n/b) + \theta(n^k \log^p n)$.

Then, we have-

$$a = \sqrt{2}$$

$$b = 2$$

$$k = 0$$

$$p = 1$$

Now, $a = \sqrt{2} = 1.414$ and $b^k = 2^0 = 1$.

Clearly, $a > b^k$.

So, we follow case-01.

So, we have-

$$T(n) = \theta(n^{\log_b a})$$

$$T(n) = \theta(n^{\log_2 \sqrt{2}})$$

$$T(n) = \theta(n^{1/2})$$

Thus,

$$\boxed{\mathbf{T(n) = \theta(\sqrt{n})}}$$

Problem-05:

Solve the following recurrence relation using Master's theorem-

$$T(n) = 8T(n/4) – n^2logn$$

Solution-

- The given recurrence relation does not correspond to the general form of Master's theorem.
- So, it can not be solved using Master's theorem.

Problem-06:

Solve the following recurrence relation using Master's theorem-

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

Solution-

- We write the given recurrence relation as $T(n) = 3T(n/3) + n$.
- This is because in the general form, we have θ for function $f(n)$ which hides constants in it.
- Now, we can easily apply Master's theorem.

We compare the given recurrence relation with $T(n) = aT(n/b) + θ (n^k log^p n)$.

Then, we have-

$$a = 3$$

$$b = 3$$

$$k = 1$$

$$p = 0$$

Now, a = 3 and $b^k = 3^1 = 3$.

Clearly, $a = b^k$.

So, we follow case-02.

Since p = 0, so we have-

$$T(n) = \theta\ (n^{\log_b a}.\log^{p+1} n)$$

$$T(n) = \theta\ (n^{\log_3 3}.\log^{0+1} n)$$

$$T(n) = \theta\ (n^1.\log^1 n)$$

Thus,

$$\boxed{\mathbf{T(n) = \theta\ (nlogn)}}$$

Problem-07:

Form a recurrence relation for the following code and solve it using Master's theorem-

```
A(n)
{
   if(n<=1)
     return 1;
   else
     return A(√n);
}
```

Solution-

- We write a recurrence relation for the given code as $T(n) = T(\sqrt{n}) + 1$.
- Here 1 = Constant time taken for comparing and returning the value.
- We can not directly apply Master's Theorem on this recurrence relation.
- This is because it does not correspond to the general form of Master's theorem.

- However, we can modify and bring it in the general form to apply Master's theorem.

Let-

$$n = 2^m \quad \ldots\ldots(1)$$

Then-

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

Now, let $T(2^m) = S(m)$, then $T(2^{m/2}) = S(m/2)$

So, we have-

$$S(m) = S(m/2) + 1$$

Now, we can easily apply Master's Theorem.

We compare the given recurrence relation with $S(m) = aS(m/b) + \theta\ (m^k \log^p m)$.

Then, we have-

$$a = 1$$

$$b = 2$$

$$k = 0$$

$$p = 0$$

Now, $a = 1$ and $b^k = 2^0 = 1$.

Clearly, $a = b^k$.

So, we follow case-02.

Since $p = 0$, so we have-

$$S(m) = \theta\ (m^{\log_b a}.\log^{p+1} m)$$

$$S(m) = \theta\ (m^{\log_2 1}.\log^{0+1} m)$$

$$S(m) = \theta\,(m^0.\log^1 m)$$

Thus,

$$S(m) = \theta(\log m) \quad ……(2)$$

Now,

- From (1), we have $n = 2^m$.
- So, $\log n = m\log 2$ which implies $m = \log_2 n$.

Substituting in (2), we get-

$$S(m) = \theta(\log\log_2 n)$$

$$\boxed{\mathbf{T(n) = \theta\,(\log\log_2 n)}}$$