

Master Method

The Master Method is used for solving the following types of recurrence

$T(n) = aT\left(\frac{n}{b}\right) + f(n)$ with $a \geq 1$ and $b \geq 1$ be constant & $f(n)$ be a function and $\frac{n}{b}$ can be interpreted as

Let $T(n)$ is defined on non-negative integers by the recurrence.

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

In the function to the analysis of a recursive algorithm, the constants and function take on the following significance:

- n is the size of the problem.
- a is the number of subproblems in the recursion.
- n/b is the size of each subproblem. (Here it is assumed that all subproblems are essentially the same size.)
- $f(n)$ is the sum of the work done outside the recursive calls, which includes the sum of dividing the problem and the sum of combining the solutions to the subproblems.
- It is not possible always bound the function according to the requirement, so we make three cases which will tell us what kind of bound we can apply on the function.

Master Theorem:

It is possible to complete an asymptotic tight bound in these three cases:

$$T(n) = \begin{cases} \Theta(n^{\log_b a}) & f(n) = O(n^{\log_b a - \epsilon}) \\ \Theta(n^{\log_b a} \log n) & f(n) = \Theta(n^{\log_b a}) \\ \Theta(f(n)) & f(n) = \Omega(n^{\log_b a + \epsilon}) \text{ AND } \\ & af(n/b) < cf(n) \text{ for large } n \end{cases} \begin{cases} \epsilon > 0 \\ c < 1 \end{cases}$$

Case1: If $f(n) = O(n^{\log_b a - \epsilon})$ for some constant $\epsilon > 0$, then it follows that:

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

Example:

$$T(n) = 8T\left(\frac{n}{2}\right) + 1000n^2 \text{ apply master theorem on it.}$$

Solution:

Compare $T(n) = 8T\left(\frac{n}{2}\right) + 1000n^2$ with

$$T(n) = aT\left(\frac{n}{b}\right) + f(n) \text{ with } a \geq 1 \text{ and } b > 1$$

$$a = 8, b = 2, f(n) = 1000n^2, \log_b a = \log_2 8 = 3$$

Put all the values in: $f(n) = O(n^{\log_b a - \epsilon})$

$$1000n^2 = O(n^{3-\epsilon})$$

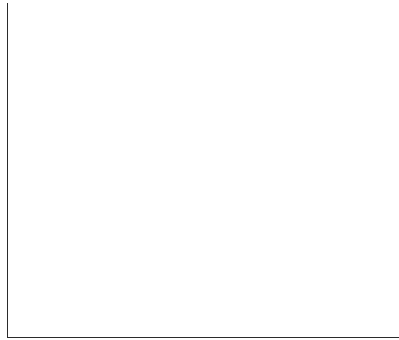
$$\text{If we choose } \epsilon = 1, \text{ we get: } 1000n^2 = O(n^{3-1}) = O(n^2)$$

Since this equation holds, the first case of the master theorem applies to the given recurrence relation, thus resulting in the conclusion:

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

$$\text{Therefore: } T(n) = \Theta(n^3)$$

Case 2: If it is true, for some constant $k \geq 0$ that:



$$f(n) = \Theta(n^{\log_b a} \log^k n) \text{ then it follows that: } T(n) = \Theta(n^{\log_b a} \log^{k+1} n)$$

Example:

$$T(n) = 2T\left(\frac{n}{2}\right) + 10n, \text{ solve the recurrence by using the master method.}$$

As compare the given problem with $T(n) = aT\left(\frac{n}{b}\right) + f(n)$ with $a \geq 1$ and $b > 1$ $a = 2, b = 2, k = 0, f(n) = 10n, \log_b a = \log_2 2 = 1$

Put all the values in $f(n) = \Theta(n^{\log_b a} \log^k n)$, we will get

$$10n = \Theta(n^1) = \Theta(n) \text{ which is true.}$$

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

Case 3: If it is true $f(n) = \Omega(n^{\log_b a + \epsilon})$ for some constant $\epsilon > 0$ and it also true that: $a f\left(\frac{n}{b}\right) \leq cf(n)$ for some constant $c < 1$ for large value of n , then :

$$T(n) = \Theta(f(n))$$

Example: Solve the recurrence relation:

$$T(n) = 2T\left(\frac{n}{2}\right) + n^2$$

Solution:

Compare the given problem with $T(n) = aT\left(\frac{n}{b}\right) + f(n)$ with $a \geq 1$ and $b > 1$

$a = 2$, $b = 2$, $f(n) = n^2$, $\log_b a = \log_2 2 = 1$

Put all the values in $f(n) = \Omega\left(n^{\log_b a + \epsilon}\right) \dots$ (Eq. 1)

If we insert all the value in (Eq.1), we will get

$n^2 = \Omega(n^{1+\epsilon})$ put $\epsilon = 1$, then the equality will hold.

$n^2 = \Omega(n^{1+1}) = \Omega(n^2)$

Now we will also check the second condition:

$$2\left(\frac{n}{2}\right)^2 \leq cn^2 \Rightarrow \frac{1}{2}n^2 \leq cn^2$$

If we will choose $c = 1/2$, it is true:

$$\frac{1}{2}n^2 \leq \frac{1}{2}n^2 \quad \forall n \geq 1$$

So it follows: $T(n) = \Theta(f(n))$

$T(n) = \Theta(n^2)$

← Prev

Next →

 [For Videos Join Our Youtube Channel: Join Now](#)


Feedback


- Send your Feedback to feedback@javatpoint.com

Help Others, Please Share




Learn Latest Tutorials

 [Splunk tutorial](#)
Splunk

 [SPSS tutorial](#)
SPSS


 [Swagger tutorial](#)
Swagger


 [T-SQL tutorial](#)
Transact-SQL


 [Tumblr tutorial](#)
Tumblr

 [React tutorial](#)
ReactJS

 [Regex tutorial](#)
Regex

 [Reinforcement learning tutorial](#)
Reinforcement Learning

 [R Programming tutorial](#)
R Programming






 [RxJS tutorial](#)
RxJS

 [React Native tutorial](#)
React Native












 [Python Design Patterns](#)
Python Design Patterns

 Python Pillow tutorial Python Pillow	 Python Turtle tutorial Python Turtle	 Keras tutorial Keras
--	--	--

Preparation

 Aptitude Aptitude	 Logical Reasoning Reasoning	 Verbal Ability Verbal Ability	 Interview Questions Interview Questions	 Company Interview Questions Company Questions
--	--	--	--	--

Trending Technologies

 Artificial Intelligence Artificial Intelligence	 AWS Tutorial AWS	 Selenium tutorial Selenium	 Cloud Computing Cloud Computing	 Hadoop tutorial Hadoop	 ReactJS Tutorial ReactJS
 Data Science Tutorial Data Science	 Angular 7 Tutorial Angular 7	 Blockchain Tutorial Blockchain	 Git Tutorial Git	 Machine Learning Tutorial Machine Learning	 DevOps Tutorial DevOps

B.Tech / MCA

 DBMS tutorial DBMS	 Data Structures tutorial Data Structures	 DAA tutorial DAA	 Operating System Operating System	 Computer Network tutorial Computer Network	 Compiler Design tutorial Compiler Design
 Computer Organization and Architecture Computer Organization	 Discrete Mathematics Tutorial Discrete Mathematics	 Ethical Hacking Ethical Hacking	 Computer Graphics Tutorial Computer Graphics	 Software Engineering Software Engineering	 html tutorial Web Technology
 Cyber Security tutorial Cyber Security	 Automata Tutorial Automata	 C Language tutorial C Programming	 C++ tutorial C++	 Java tutorial Java	 .Net Framework tutorial .Net
 Python tutorial Python	 List of Programs Programs	 Control Systems tutorial Control System	 Data Mining Tutorial Data Mining	 Data Warehouse Tutorial Data Warehouse	

